UNIVERSITE SULTAN MOULAY SLIMANE
ECOLE NATIONALE DES SCIENCES APPLIQUEES
KHOURIBGA

Rapport de projet de fin d'année

Master Big Data et Aide à la Décision

Technologies d'apprentissage automatique et de Blockchain pour la sécurité de l'IdO

Soutenu le 09/10/2023 devant la commission d'examen composée de :

Président	Pr	.....
Examinateur	Pr	.....
Encadrant	Mr.HAFIDI Imad	ENSA
co-superviseur	Mme.TABBAA Hiba	ENSA

Remerciement

Après avoir rendu grâce à Dieu le tout puissant et le miséricordieux, nous tenons à exprimer toute notre reconnaissance à notre encadrant, Mr Pr.Imad HAFIDI et notre co-encadrante Mme Hiba TABBAA, nous les remercions de nous avoir orienté, aidé et conseillé.
Nous adressons nos sincères remerciements à tous les professeurs, intervenants et toutes les personnes qui par leurs paroles, leurs conseils et leurs critiques ont guidé nos réflexions.
Nous remercions nos très chers parents, qui ont toujours là pour nous, nous désirons aussi remercier nos amis pour leur soutien inconditionnel et leurs encouragements tout au long de notre démarche.

ملخص

تطورت تكنولوجيا انترنت الأشياء بشكل كبير، مما سمح للأجهزة بالتواصل وتبادل البيانات، ولكن مع تقدمها، ظهرت تحديات أمنية تهدد الأمان والحماية. تصاحب هذه التكنولوجيا مخاطر الاختراق والتهديدات السيبرانية، مما يستدعي البحث عن حلول مبتكرة.

من هنا، يأتي دور دمج تكنولوجيا البلوكشين والذكاء الاصطناعي لتحقيق الحماية والأمان في مجال انترنت الأشياء. تقنية البلوكشين تقدم آليات أمان تعتمد على التشفير والتدقيق اللامركزي، مما يسهم في ضمان مصداقية البيانات والمعلومات. ومن جهة أخرى، يستفيد الذكاء الاصطناعي من تقنيات التعلم العميق والكشف التلقائي لاكتشاف السلوكيات غير العادية والتهديدات الأمنية.

هذا الدمج بين تكنولوجيا البلوكشين والذكاء الاصطناعي يشكل تقدمًا مهمًا في تعزيز أمان إنترنت الأشياء وحمايتها من التهديدات. يعمل هذا الجهد المشترك على تحقيق توازن بين التطور التكنولوجي والحماية، مما يساهم في تحقيق نمو أكثر ثقة في مجال إنترنت الأشياء.

كلمات مفاتيح
انترنت الأشياء ، الذكاء الاصطناعي ، البلوكشين.

Résumé

Les réseaux IdO (Internet des objets) se sont imposés comme un nouveau paradigme informatique au cours des cinq dernières années. Ces réseaux ont réussi à s'intégrer dans les schémas d'infrastructure industrielle, se positionnant comme des dispositifs qui communiquent des informations hautement classifiées pour les entreprises les plus critiques des nations. Actuellement, et afin de trouver des alternatives pour atténuer ce risque, des solutions basées sur des algorithmes Blockchain ont été mises en place pour le transfert précis d'informations. D'autre part, des techniques d'apprentissage automatique ont été utilisées afin d'identifier d'éventuelles menaces contre les réseaux IdO.
Ce projet cherchait à intégrer les solutions précédentes pour créer un mécanisme de protection complète des réseaux d'appareils IdO, qui permettrait d'identifier les menaces et activer des mécanismes sécurisés de transfert d'informations.

Mots-clés :
IdO : Internet des objets, Blockchain, Apprentissage automatique.

Abstarct

Technology has become an indispensable part of human life, especially with the growth of the Internet of Things (IoT), which allows communication and interaction with various devices. However, IoT has been proven to be susceptible to security breaches. Therefore, it is necessary to develop foolproof solutions by creating new technologies or combining existing technologies to address security issues.
Currently, and in order to find alternatives to mitigate this risk, solutions based on Blockchain algorithms have been put in place for the accurate transfer of information. On the other hand, Machine learning have been used to identify potential threats to IoT networks.
This project sought to integrate previous solutions to create a comprehensive protection mechanism for IoT device networks, which would identify threats and enable secure information transfer mechanisms.

kyewords :
IoT, Blockchain, machine learning

Table des matières

Table des figures

Liste des tableaux

Introduction

L'Internet des Objets (IdO) est une technologie émergente et prometteuse qui automatise les processus commerciaux et académiques sous forme d'opérations simples et faciles. Son concept fondamental réside dans la connexion d'un grand nombre de dispositifs intelligents pour échanger des services et des données via Internet, sans nécessiter d'intervention humaine. Les réseaux IdO nous permettent de coopérer avec ces dispositifs sans nous impliquer directement. L'objectif principal de l'IdO est de simplifier les opérations humaines grâce à des applications intelligentes.

L'IdO est devenu le plus vaste réseau au monde, regroupant des millions de dispositifs interconnectés, tels que des voitures, des appareils mobiles, des ordinateurs portables, des bâtiments et même des vêtements équipés de capteurs. Ces capteurs transforment le monde physique en données numériques. Du fait de la structure complexe des réseaux IdO, l'hétérogénéité et la décentralisation en sont les caractéristiques clés. L'éventail des applications IdO s'élargit considérablement dans notre vie quotidienne, créant de nouvelles opportunités communautaires grâce à la croissance massive de ces réseaux.

Bien que l'IdO ait prouvé ses avantages dans plusieurs domaines, il présente de nombreux défis, notamment la confiance, la sécurité et l'identification des objets, en particulier compte tenu des ressources limitées des dispositifs IdO. Ces défis, qui existent également dans d'autres environnements, peuvent avoir un impact plus préjudiciable sur les réseaux IdO, ouvrant ainsi la voie aux intrusions. Les principales limites des réseaux IdO résident dans les capacités limitées des systèmes d'exploitation en termes de puissance, de stockage et de capacités de calcul limitées des dispositifs finaux. Par conséquent, peu de dispositifs IdO peuvent intégrer les mécanismes de sécurité de base, ce qui rend les exigences de sécurité difficiles à garantir. De plus, la connectivité, la visualisation et l'analyse des données posent des problèmes, notamment en raison de la nature hétérogène de l'IdO et des protocoles de réseau peu fiables. Il est donc un véritable défi de fusionner diverses technologies et dispositifs au sein d'un seul réseau de manière sécurisée.

En raison de ces limites, l'IdO présente de nombreuses vulnérabilités en matière de sécurité qui peuvent se transformer en véritables menaces. De plus, ces limitations rendent les réseaux IdO vulnérables aux attaques de cybersécurité contemporaines utilisant des approches techniques avancées. Pour faire face efficacement aux limitations de l'IdO, de nombreuses études en matière de sécurité ont été menées pour renforcer la sécurité de l'IdO en utilisant d'autres technologies émergentes. D'après les études récentes, l'apprentissage

automatique et la Blockchain sont les principaux vecteurs de transformation de l'IdO et de résolution de ses problèmes. De plus, les approches d'apprentissage automatique bio-inspirées sont devenues un modèle approprié pour les réseaux IdO dynamiques et évolutifs en raison des opérations communes entre les phénomènes biologiques et l'IdO.

Pour surmonter ces limitations, plusieurs études de sécurité ont été menées pour renforcer la sécurité de l'IdO grâce à l'intégration de technologies émergentes. L'apprentissage automatique et la Blockchain, sans oublier les approches d'apprentissage automatique bio-inspirées, sont des solutions clés pour transformer l'IdO et relever ces défis. Les modèles basés sur l'apprentissage automatique peuvent générer une "Intelligence des Objets" reposant sur une détection centralisée, tandis que la Blockchain, en tant que technologie de décentralisation, offre un registre distribué garantissant le partage sécurisé des données de manière inviolable, résolvant ainsi les problèmes de confidentialité et de fiabilité.

La convergence de l'IdO, de l'apprentissage automatique et de la Blockchain promet de nouvelles avancées. La Blockchain renforce la confidentialité et la fiabilité, tandis que l'apprentissage automatique peut créer des algorithmes analytiques avancés pour défendre l'IdO contre les cyberattaques.

Ce rapport est divisé en trois chapitres distincts : le premier aborde l'état de l'art, le deuxième présente la méthodologie de notre recherche, et enfin, les résultats des algorithmes utilisés sont présentés dans le troisième chapitre du rapport.

Chapitre 1 ETAT DE L'ART

1.1 Introduction

L'Internet des Objets (IdO) est une technologie en constante expansion qui révolutionne de nombreux secteurs, notamment la santé, la sécurité, l'agriculture et le transport. Elle permet aux objets physiques de collecter, traiter et échanger des données en temps réel, ouvrant ainsi la voie à l'automatisation et à la prise de décision intelligente.

La blockchain, en tant que grand livre numérique distribué, a révolutionné divers domaines en permettant des échanges décentralisés, notamment dans le secteur financier, où elle a facilité la collecte de fonds directement auprès des investisseurs. L'intégration de l'Internet des Objets (IdO) avec la blockchain présente un potentiel considérable en sécurisant les données de l'IdO grâce à l'immuabilité de la blockchain, révolutionnant ainsi le financement de projets innovants tout en préservant la confidentialité des données de l'IdO.

Parallèlement, l'apprentissage automatique, une technologie émergente, renforce la sécurité de l'IdO en détectant les anomalies dans les flux de données de l'IdO et en prévenant les incidents de sécurité. Cette convergence de l'IdO, de la blockchain et de l'apprentissage automatique ouvre la voie à un avenir où les systèmes seront plus intelligents, plus sécurisés et plus performants. Ces avancées technologiques transforment la manière dont nous interagissons avec le monde connecté qui nous entoure, offrant ainsi de nouvelles perspectives passionnantes pour l'innovation et le progrès.

1.2 Internet des Objets

1.2.1 Définition

Selon l'Union internationale des télécommunications, l'Internet des objets (IdO) est une « infrastructure mondiale pour la société de l'information, qui permet de disposer de services évolués en interconnectant des objets (physiques ou virtuels) grâce aux techno-

logies de l'information et de la communication interopérables existantes ou en évolution ».

1.2.2 Architecture de l'IdO

L'Internet des Objets (IdO) est un réseau de dispositifs physiques, de véhicules, d'objets connectés et autres, qui sont intégrés avec des capteurs, des logiciels et des technologies de communication pour collecter et échanger des données. Voici les quatre étapes clés de l'architecture de l'IdO :

-Les capteurs et les dispositifs : la première étape de l'IdO consiste à intégrer des capteurs et des dispositifs physiques à l'environnement. Ces capteurs peuvent être utilisés pour collecter des données sur la température, l'humidité, la pression, l'emplacement, etc.

-La connectivité : une fois que les capteurs et les dispositifs sont en place, il est nécessaire de les connecter au réseau. Les technologies de communication comme le Wi-Fi, Bluetooth, Zigbee, LTE, etc. sont utilisées pour connecter les dispositifs à l'Internet.

-Le Cloud computing : les données collectées par les capteurs et les dispositifs sont stockées et traitées dans le Cloud. Les plateformes IdO peuvent être utilisées pour gérer ces données, les analyser et les visualiser.

-L'analyse et l'interaction : la dernière étape de l'IdO consiste à utiliser les données collectées pour prendre des décisions. Les données sont analysées et utilisées pour fournir des informations qui peuvent être utilisées pour améliorer les processus et les performances. Les dispositifs peuvent également être programmés pour interagir avec l'environnement en fonction des données collectées[1].

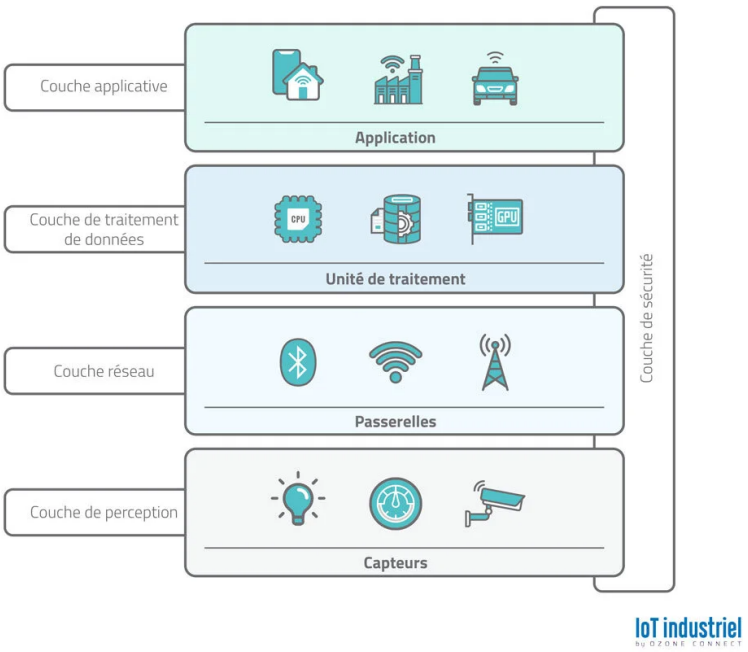


FIG. 1.1 – architecture IdO à quatre couches[2].

1.2.3 Attaques visant l'Internet des Objets

Les attaques contre l'Internet des Objets (IdO) se divisent en deux catégories principales :
les attaques passives et les attaques actives. Les attaques passives impliquent généralement l'interception de données en transit entre les objets connectés et les serveurs, pouvant compromettre des informations sensibles telles que les mots de passe ou les numéros de carte de crédit. Les attaques passives incluent le "sniffing"
. En revanche, les attaques actives visent à manipuler ou altérer les données en transit, prenant diverses formes, notamment les attaques par déni de service (DDoS) et les attaques de type "commande et contrôle", où les pirates prennent le contrôle à distance des objets connectés à des fins malveillantes.

Ces attaques peuvent avoir des conséquences graves pour la vie privée et la sécurité des utilisateurs, d'où la nécessité de mettre en place des mesures de sécurité telles que la surveillance, la détection des attaques et des procédures de sécurité pour les objets connectés.
Voici quelques exemples d'attaques visant l'Internet des Objets (IdO) :
◦ Attaques passives :

- Attaques de type "sniffing" : interception du trafic réseau entre les objets connectés et les serveurs pour récupérer des données sensibles.

- Attaques par injection de code : insertion de code malveillant dans les données transitant entre les objets connectés et les serveurs, ce qui permet aux pirates de prendre le contrôle des objets ou d'intercepter des données.

◦ Attaques actives :

- Attaques par déni de service (DDoS) : utilisation d'un grand nombre d'objets connectés pour saturer le réseau et rendre le service indisponible.

- Attaques de type "commande et contrôle" : prise de contrôle à distance des objets connectés pour les utiliser à des fins malveillantes, telles que la collecte d'informations personnelles ou la propagation de logiciels malveillants.

- Attaques par ransomware : les objets connectés sont infectés par un logiciel malveillant qui chiffre les données et demande une rançon pour les déchiffrer.

- Attaques physiques : les objets connectés peuvent être physiquement endommagés pour empêcher leur fonctionnement normal. - Attaques par botnets : des botnets sont des réseaux d'objets connectés infectés par un logiciel malveillant qui permet aux pirates de les contrôler à distance. Les botnets sont souvent utilisés pour mener des attaques par déni de service (DDoS) massives.

- Attaques par sniffing Bluetooth : les objets connectés peuvent utiliser Bluetooth pour communiquer avec d'autres dispositifs. Les pirates peuvent intercepter les données échangées par les objets connectés à l'aide d'outils de sniffing Bluetooth.

- Attaques de type "Man-in-the-Middle" : les pirates peuvent utiliser des techniques de type "Man-in-the-Middle" pour intercepter les communications entre les objets connectés

et les serveurs, et ainsi capturer des données sensibles.

- Attaques par falsification de données : les pirates peuvent altérer les données envoyées par les objets connectés aux serveurs pour obtenir des résultats indésirables, tels que la modification de la température d'un thermostat ou l'ouverture d'une porte de garage.

- Attaques par exploitation de vulnérabilités : les objets connectés peuvent contenir des vulnérabilités de sécurité qui peuvent être exploitées par les pirates pour prendre le contrôle de l'objet ou pour voler des données.[3]

Le tableau suivant résume les différentes attaques citées ci-dessus :

Nom de l'attaque	But et résultat de l'attaque	Menace	Active ou Passive
Dos	-Saturer un serveur ou bloquer le trafic. -Rendre un service non disponible	-Intégrité. -Disponibilité. -Confidentialité.	Active
Man-in-the-Middle	-Intercepter les communications entre deux parties et contrôler la conversation. -Ecouter, modifier ou supprimer des données.	-Intégrité. -Confidentialité.	Active
L'usurpation d'identité	-Vol d'identité.	-Confidentialité. -Authentification.	Active
Lecture du message	-Ecouter et lire les données envoyées entre deux entités.	-Confidentialité.	Passive
Analyse du trafic	-Analyse des informations concernant les données transmises.	-Confidentialité.	Passive
Rejeu	-Ecouter, modifier ou supprimer des données.	-Intégrité. -Confidentialité. -Disponibilité.	Active
Modification des messages	- Modifier, retarder, réordonner des données.	-Intégrité. -Disponibilité.	Active

TABLE 1.1 – Les types d'attaques dans l'IdO[3].

1.2.4 Services de sécurité dans IdO

Les services de sécurité dans l'Internet des Objets (IdO) sont des solutions qui visent à protéger les objets connectés et les réseaux IdO contre les attaques malveillantes. Voici quelques-uns des services de sécurité couramment utilisés dans l'IdO :

-Identification et authentification des objets connectés : cela implique l'attribution d'identifiants uniques à chaque objet connecté et la mise en place de mécanismes d'authentification pour garantir que seuls les objets connectés autorisés peuvent accéder au réseau.

-Cryptographie : la cryptographie est utilisée pour chiffrer les données échangées entre les objets connectés et les serveurs, empêchant ainsi les pirates de lire ou d'altérer les données en transit.

-Surveillance des comportements anormaux : la surveillance des comportements anormaux sur le réseau peut aider à détecter les attaques et à prendre des mesures préventives.

-Mise à jour des logiciels : les mises à jour régulières du logiciel des objets connectés peuvent corriger les vulnérabilités de sécurité et empêcher les pirates d'exploiter les failles de sécurité.

-Pare-feu : la mise en place d'un pare-feu peut empêcher les pirates d'accéder au réseau ou de prendre le contrôle des objets connectés.

-Gestion des accès : la gestion des accès peut être utilisée pour contrôler l'accès aux données et aux ressources des objets connectés, en permettant uniquement aux utilisateurs autorisés d'accéder aux informations sensibles.

-Sécurité physique : la sécurité physique peut être utilisée pour protéger les objets connectés contre les attaques physiques, telles que le vol ou l'endommagement.

Ces services de sécurité sont essentiels pour protéger les objets connectés et les réseaux IdO contre les menaces de sécurité croissantes. Il est important de mettre en place une stratégie de sécurité complète pour garantir la sécurité de l'IdO[4].

1.3 Utilisation de la Blockchain pour renforcer la sécurité de l'IdO

La blockchain, avancée majeure, instaure un système sécurisé et décentralisé, révolutionnant la gestion des données numériques. Cette structure, constituée de blocs intègres, garantit la transparence et la traçabilité, ouvrant la voie à des applications novatrices. En convergence avec l'Internet des Objets (IdO), une percée majeure, cette alliance devient puissante.

L'IdO, liant dispositifs physiques via un réseau, favorise l'échange de données et la collaboration entre machines. Intégrant la sécurité de la blockchain et les capacités interconnectées de l'IdO, cette fusion ouvre des opportunités captivantes dans divers domaines.

Dans cette étude, nous explorons comment la blockchain et l'Internet des Objets fusionnent pour optimiser leurs avantages, tout en fournissant des solutions novatrices pour les défis majeurs, notamment la consolidation de la sécurité dans l'IdO.

1.3.1 Définition

La blockchain est un registre partagé et immuable qui facilite le processus d'enregistrement des transactions et de suivi des actifs dans un réseau d'entreprise. Un actif peut être tangible (une maison, une voiture, de l'argent, un terrain) ou incorporel (propriété intellectuelle, brevets, droits d'auteur, image de marque). Pratiquement tout ce qui a de la valeur peut être suivi et échangé sur un réseau blockchain, ce qui réduit les risques et les coûts pour toutes les personnes impliquées.

1.3.2 Architecture de la Blockchain

L'architecture de la blockchain est fondée sur un réseau décentralisé où les données sont enregistrées sous forme de blocs reliés les uns aux autres de manière chronologique, créant ainsi une chaîne de blocs. Chaque bloc contient des données, comme des transactions, et est identifié par un hachage cryptographique unique, assurant l'intégrité de la chaîne. Les principaux éléments de l'architecture sont la décentralisation, où de nombreux nœuds répartis dans le réseau participent à la validation et au stockage des données, la transparence, car la chaîne est accessible à tous les participants, et l'immutabilité, car les données enregistrées dans un bloc deviennent immuables une fois validées. L'architecture de la blockchain permet ainsi de créer un système sécurisé, résistant à la censure, et offre une gamme d'applications allant des cryptomonnaies aux contrats intelligents, sans nécessiter de tiers de confiance.

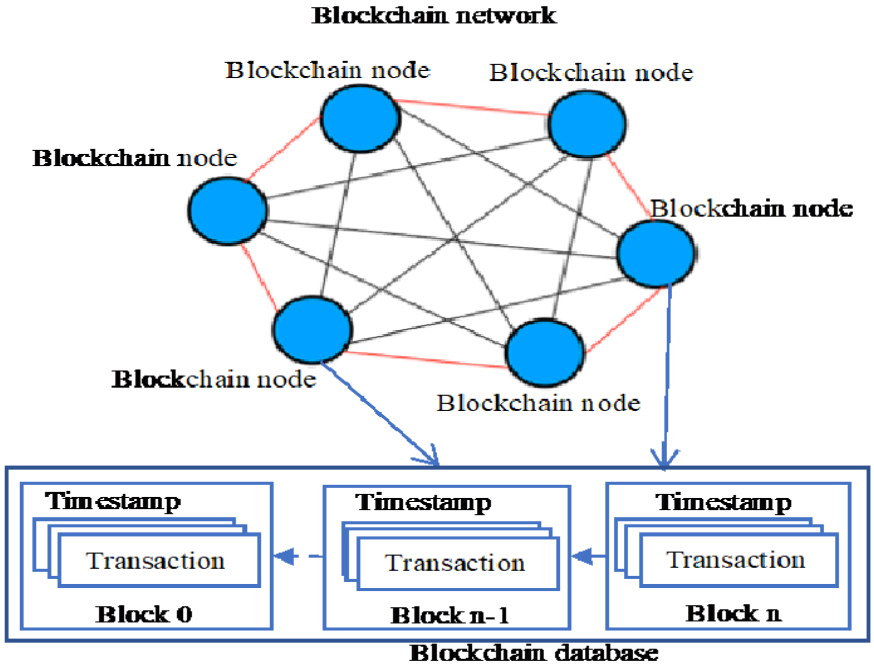


FIG. 1.2 – Architecture de la Blockchain[5].

1.3.2.1 Composants de Blockchain

La blockchain est constituée de plusieurs composantes clés qui travaillent ensemble pour créer son fonctionnement caractéristique. Voici les principales composantes d'une blockchain :

Transaction : Les transactions sont des événements sécurisés cryptographiquement et vérifiés, assurant la sécurité et l'authenticité des données. L'immuabilité des transactions renforce leur validité.

Bloc : Un bloc est une unité fondamentale de la blockchain, contenant un ensemble de transactions et d'autres données. Chaque bloc est relié au précédent et au suivant par des mécanismes cryptographiques.

Chaine : Les blocs sont organisés en une séquence linéaire, créant ainsi une chaîne de blocs. Chaque bloc contient une référence au bloc précédent, ce qui assure la continuité et l'intégrité de la chaîne.

Hachage : Un hachage est une empreinte numérique unique générée à partir des données d'un bloc. Cela permet d'identifier rapidement le contenu du bloc et garantit que même la plus petite modification des données entraînera un hachage complètement différent.

Algorithmes de Consensus : Les protocoles de consensus sont utilisés dans les systèmes distribués, pour garantir que tous les nœuds du réseau s'accordent sur l'état actuel du système. Ces protocoles sont responsables d'obtenir un accord distribué entre les nœuds.

- Preuve de Travail (PoW) : Utilisé par Bitcoin, il nécessite une résolution de problèmes mathématiques pour ajouter des blocs, garantissant la sécurité mais nécessitant beaucoup d'énergie.
- Preuve d'Enjeu (PoS) : Employé par des blockchains comme Ethereum 2.0, il sélectionne les validateurs en fonction de la quantité de cryptomonnaie qu'ils "mettent en jeu".
- Preuve d'Autorité (PoA) : Utilisé dans certaines blockchains privées, il exige que les validateurs prouvent leur identité pour maintenir l'intégrité du réseau.
- Practical Byzantine Fault Tolerance (PBFT) : Employé dans des blockchains privées, il garantit un accord entre les nœuds, même en présence de nœuds défaillants.

les contrats intelligents : Les contrats intelligents sont des programmes autonomes qui s'exécutent automatiquement lorsque certaines conditions sont remplies. Ils permettent l'automatisation des processus et des transactions dans la blockchain.

type de blockchain : Les blockchains se divisent en deux catégories majeures. Les blockchains publiques, telles que Bitcoin et Ethereum, sont accessibles à tous, décentralisées et sécurisées par la preuve de travail. Les blockchains privées sont restreintes à un groupe spécifique de participants, souvent gérées par une entité centrale, et peuvent utiliser des mécanismes de consensus différents. Il existe également des blockchains hybrides qui fusionnent des caractéristiques des deux types pour répondre à des besoins spécifiques.

1.3.3 Processus de la Blockchain

Le processus de la blockchain comprend plusieurs étapes clés :

- Création de la transaction : L'utilisateur A crée une transaction indiquant qu'il souhaite envoyer une certaine quantité de crypto-monnaie ou un actif numérique à l'utilisateur B. Cette transaction est composée d'informations telles que l'adresse du destinataire, le montant de la transaction et une signature numérique pour prouver l'authenticité de la demande.

- Diffusion de la transaction : Une fois la transaction créée, elle est diffusée sur le réseau de la blockchain. Tous les nœuds du réseau reçoivent la transaction et la vérifient pour s'assurer qu'elle est valide et conforme aux règles du protocole de la blockchain.

- Validation par les mineurs : Les mineurs du réseau rassemblent un groupe de transactions valides (généralement appelé "mempool") et commencent à résoudre un problème mathématique complexe, généralement une preuve de travail, pour créer un nouveau bloc.
- Ajout de la transaction au bloc : Lorsqu'un mineur résout le problème et crée un nouveau bloc, la transaction entre A et B est incluse dans ce bloc. Le bloc est ensuite ajouté à la chaîne de blocs existante.

- Confirmation de la transaction : Une fois que la transaction est incluse dans un bloc, elle est considérée comme confirmée. Le nombre de confirmations requis peut varier en fonction de la politique de la blockchain ou de la sensibilité de la transaction.

- Réplication sur le réseau : Le nouveau bloc contenant la transaction est répliqué sur tous les nœuds du réseau, garantissant que chaque participant dispose de la même version mise à jour de la blockchain.

- Finalisation de la transaction : Une fois la transaction confirmée et ajoutée à la blockchain, la propriété de la crypto-monnaie ou de l'actif numérique est transférée de A à B de manière irréversible et sécurisée.

Ce processus décentralisé et sécurisé permet d'avoir confiance dans les données stockées dans la blockchain sans avoir besoin d'une autorité centrale de confiance. Il est largement utilisé dans le domaine des cryptomonnaies, mais ses applications s'étendent également à de nombreux autres domaines, y compris la gestion des contrats, la traçabilité des chaînes d'approvisionnement, la gestion des identités et bien plus encore.

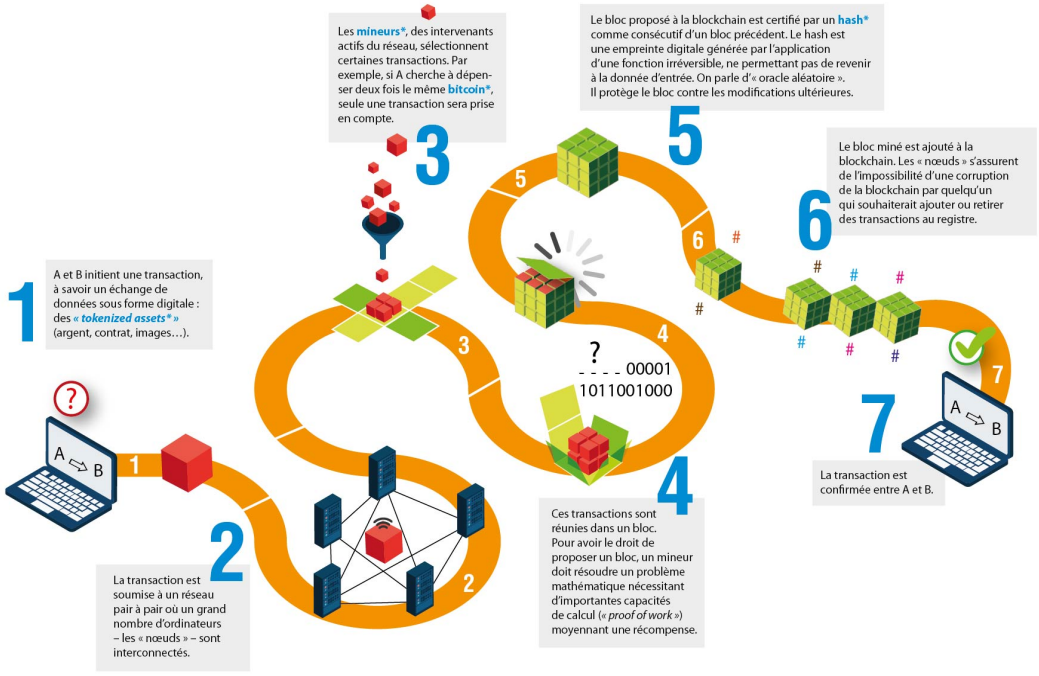


FIG. 1.3 – Processus de la Blockchain[5].

1.3.4 Application de la BC

La blockchain trouve des applications dans divers domaines en raison de ses caractéristiques de décentralisation, de sécurité et d'immuabilité. Voici quelques-unes des principales applications de la blockchain :

- Cryptomonnaies : La blockchain sert de base pour enregistrer et vérifier les transactions de manière sécurisée et décentralisée, formant ainsi le fondement des cryptomonnaies telles que Bitcoin et Ethereum.
- Logistique et chaîne d'approvisionnement : La blockchain permet de suivre et d'authentifier les produits tout au long de leur parcours, améliorant ainsi la transparence et l'efficacité de la chaîne d'approvisionnement.
- Santé : La blockchain est utilisée pour stocker en toute sécurité les dossiers médicaux des patients, garantissant la confidentialité et l'intégrité des données tout en facilitant le partage d'informations entre les professionnels de la santé.
- Identité numérique : La blockchain permet aux individus de contrôler et de sécuriser leurs données personnelles, réduisant ainsi les risques de vol d'identité et de fraude.
- Énergie : La blockchain facilite le suivi et la gestion des échanges d'énergie entre les utilisateurs, favorisant l'émergence de réseaux énergétiques décentralisés. À mesure que cette technologie continue de se développer, elle doit faire face à certains défis pour une adoption plus large dans divers secteurs. Les progrès de la blockchain nécessitent de surmonter ces obstacles afin de permettre une utilisation étendue dans différents domaines

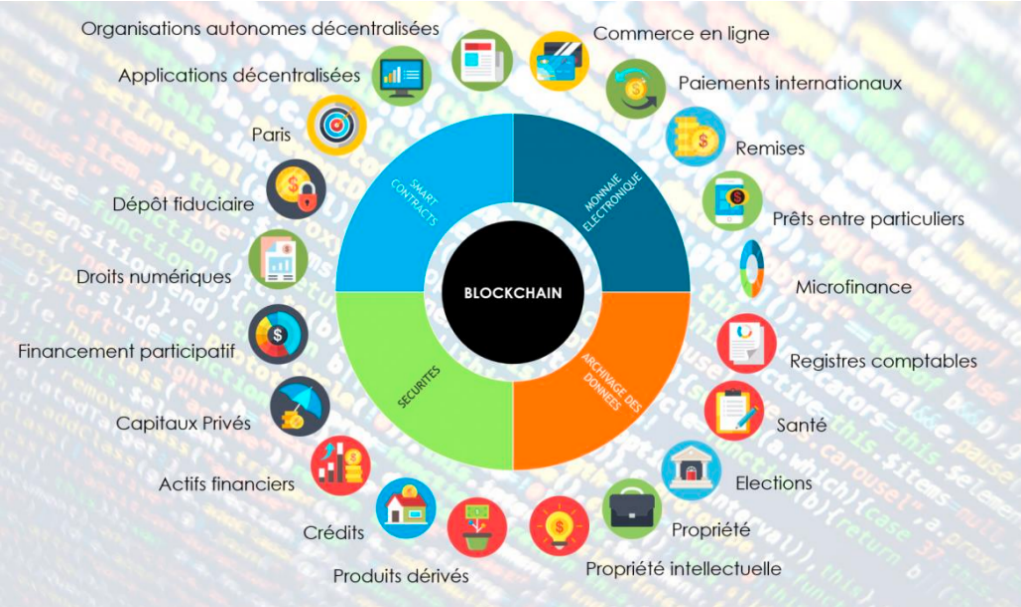


FIG. 1.4 – Applications de la Blockchain[5].

1.3.5 Blockchain pour l'IdO

Si l'on fait abstraction de cas exceptionnels, l'utilisation de la blockchain se fait sous trois formes : en tant que base de données de stockage décentralisée, en tant que registre distribué, ou en tant que support pour les services distribués fournis par des contrats intelligents. La blockchain est intégrée aux calculs de rosée et de cloudlet pour répondre aux exigences fondamentales de l'IdO, qui sont : le transfert de calcul, le stockage externe des données et la gestion du trafic réseau. Dans ce qui suit, nous présentons ces trois couches :

Couche des appareils : Cette couche, située à la périphérie du réseau, est composée d'appareils de détection et d'actionnement IdO. Ces dispositifs surveillent et contrôlent diverses applications intelligentes et envoient les données générées localement à la couche de rosée. Les appareils IdO participent à la blockchain via des serveurs capables dans les couches supérieures de rosée et de cloudlet, permettant ainsi de décharger les opérations lourdes vers ces serveurs.

Couche de rosée (dew layer) : La couche de rosée reçoit les données brutes générées par les appareils IdO et est constituée de contrôleurs plus performants connectés via la technologie blockchain. Chaque contrôleur de rosée représente un nœud dans une blockchain de consortium et couvre une petite communauté d'appareils associés. La couche de rosée est responsable de la livraison de services en temps opportun, de l'analyse des données et du traitement des données. Elle offre une localisation précise et fournit des capacités d'auto-organisation dans un environnement dynamique, évitant ainsi les risques de défaillance d'un point unique. La blockchain utilisée au niveau de rosée permet une distribution décentralisée des ressources et l'utilisation de contrats intelligents pour une meilleure efficacité et une plus grande sécurité.

Couche de cloudlet : La couche de cloudlet est composée de ressources plus puissantes, qui fournissent le traitement, l'analyse et le stockage à long terme des données, ainsi qu'une communication et une gestion à un niveau supérieur. Ces ressources de cloudlet sont configurées comme des nœuds de blockchain, permettant ainsi de participer au processus de minage pour garantir la confidentialité et l'intégrité des données. La couche de cloudlet héberge des installations de stockage et de calcul massives, maintenant une réplication complète de tous les enregistrements partagés entre elles. La blockchain utilisée à ce niveau permet une distribution sécurisée, évolutive et fiable des services à faible coût, avec une haute disponibilité et un accès à la demande. Les contrats intelligents sont également utilisés pour réduire la latence et augmenter le débit des ressources distribuées au sein de la couche de cloudlet.

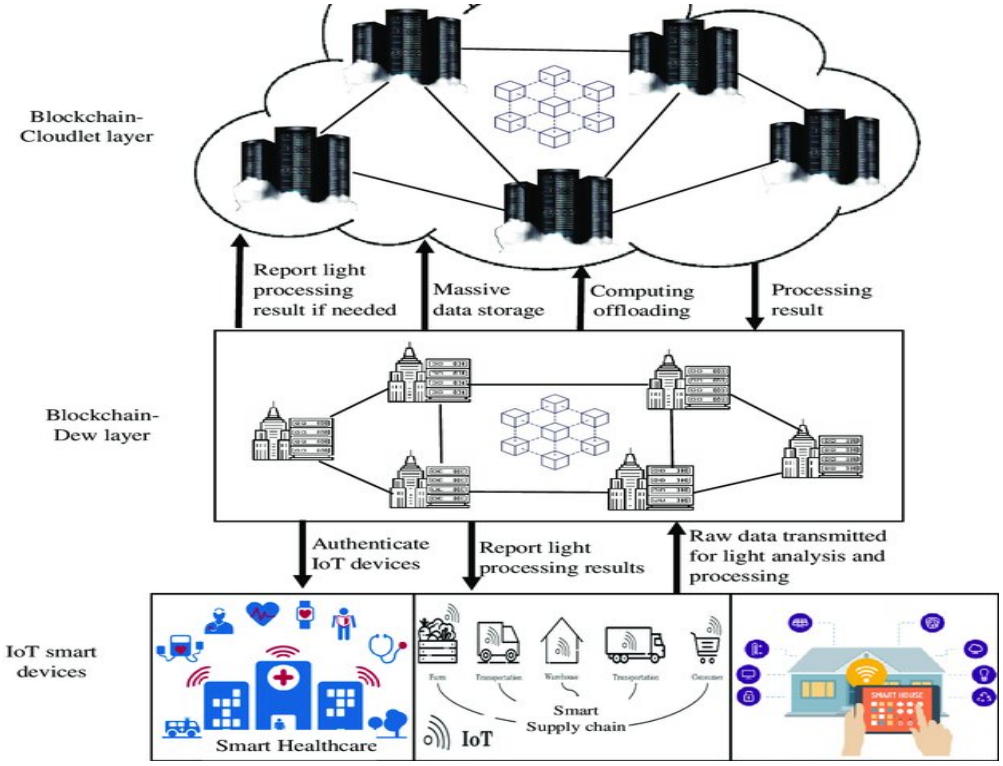


FIG. 1.5 – Architecture de la Blockchain intégrée en IdO[6].

1.3.6 Plateformes et Frameworks Blockchain

La présente section aborde le concept de plateformes et de frameworks blockchain, essentiels pour la conception et le déploiement d'applications exploitant la technologie blockchain. Ces solutions fournissent une infrastructure solide et des outils nécessaires pour développer des applications décentralisées, améliorant la sécurité, la transparence et l'intégrité des données dans divers contextes. Dans cette section, nous examinerons certaines des plateformes et des frameworks blockchain notables.

1. Ethereum : Ethereum, une figure éminente dans le domaine, permet la création de contrats intelligents et d'applications décentralisées (DApps). Basé sur un modèle de machine virtuelle décentralisée (EVM), Ethereum offre un langage de programmation natif, Solidity, pour développer des contrats intelligents.
2. Hyperledger Fabric : Hyperledger Fabric, au sein du projet Hyperledger, se démarque comme une plateforme blockchain à permissions conçue pour les cas d'utilisation en entreprise. Elle se distingue par son architecture modulaire, qui permet aux organisations de personnaliser les mécanismes de consensus et les fonctionnalités de confidentialité, tout en prenant en charge les canaux privés.
3. Hyperledger Sawtooth : Autre projet de la famille Hyperledger, Hyperledger Sawtooth se démarque par sa modularité et sa scalabilité. Cette plateforme offre une flexibilité exceptionnelle grâce à son architecture modulaire, permettant l'expérimentation de divers algorithmes de consensus et de langages de contrats intelligents.
4. Corda : Corda s'adresse spécifiquement aux entreprises et aux institutions financières en mettant l'accent sur la confidentialité et la sécurité. Son approche unique du consensus et son orientation vers la validation par les parties prenantes en font une option intéressante pour les cas où la confidentialité est primordiale.
5. EOSIO : EOSIO se démarque par sa capacité à fournir des applications décentralisées rapides et évolutives. Basée sur un mécanisme de consensus délégué (DPoS), EOSIO supporte plusieurs langages de programmation pour la création de contrats intelligents.
6. Truffle : Truffle est un framework de développement dédié à Ethereum. Il fournit une suite d'outils pour faciliter le développement, le test et le déploiement de contrats intelligents.
7. Quorum : Quorum, basé sur Ethereum, se distingue par sa focalisation sur les besoins de l'entreprise. En mettant l'accent sur la confidentialité et les contrats privés, Quorum s'aligne avec les cas d'utilisation spécifiques au secteur financier.
8. Binance Smart Chain : Binance Smart Chain, développée par Binance, permet de développer des applications compatibles avec Ethereum. Elle offre une alternative pour les cas où la compatibilité avec l'EVM est essentielle.

Ces plateformes et frameworks blockchain offrent des fonctionnalités distinctes pour répondre à une variété de cas d'utilisation. Le choix de la plateforme appropriée dépendra des exigences spécifiques de chaque projet, en tenant compte des facteurs tels que la confidentialité, la scalabilité et le contexte industriel.

1.4 Utilisation de l'apprentissage automatique pour renforcer la sécurité de l'IdO

Le but de cette section est d'explorer comment l'apprentissage automatique peut être appliqué pour renforcer la sécurité de l'Internet des Objets (IdO). Nous examinerons comment l'apprentissage automatique peut jouer un rôle important dans la détection précoce d'incidents et dans la réponse rapide aux menaces, offrant ainsi une perspective approfondie sur son potentiel pour garantir la protection des systèmes IdO.

1.5 Apprentissage automatique pour la sécurité de l'IdO

1.5.1 Apprentissage automatique

L'intelligence artificielle (IA) est un domaine de l'informatique qui se concentre sur la création de machines qui peuvent imiter ou surpasser les capacités humaines dans des tâches intelligentes. L'IA peut être appliquée à une grande variété de domaines, notamment la reconnaissance de formes, la prise de décision, la compréhension du langage naturel, la robotique, la vision par ordinateur et bien plus encore.
Le domaine de l'intelligence artificielle est scindé en plusieurs sous domaines imbriqués.

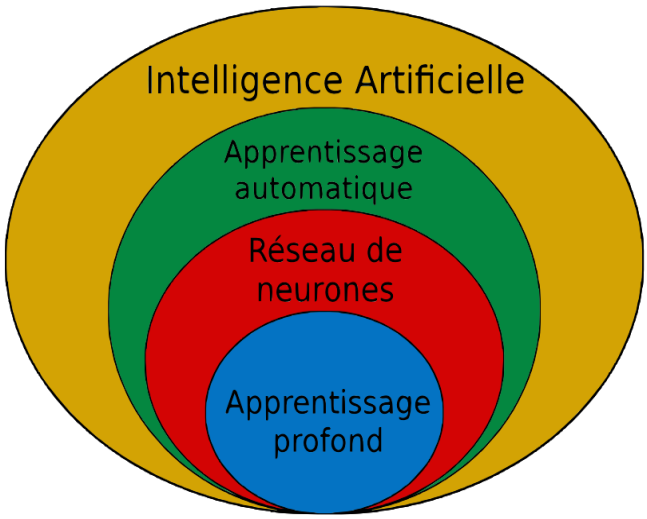


Fig. 1.6 – Schéma de décomposition du domaine de l'IA et de ces sous-domaines [7].

L'apprentissage automatique est défini par Arthur Samuel comme "le domaine d'étude qui fournit aux ordinateurs la capacité d'apprendre sans être explicitement programmés". Cette discipline, également connue sous le nom de "machine learning", est un domaine d'étude de l'intelligence artificielle qui, à partir de bases de données, lance des algorithmes, pour obtenir des analyses prédictives à des fins précises et établit des corrélations entre

divers événements.

Le mode opératoire de l'apprentissage automatique est divisé en deux phases. Dans la première, la phase « d'apprentissage », une partie des données est apprise ; dans la seconde, la phase de « vérification », la seconde partie des données est utilisée. De plus, pour comprendre l'apprentissage automatique, il est nécessaire d'expliquer également les termes « apprentissage supervisé » et « apprentissage non supervisé ». Le premier, construit un modèle de corrélation entre variables connues, cependant, le second, prend en compte un ensemble de variables et extrait les corrélations les plus fortes. L'apprentissage supervisé est considéré comme subjectif, puisque c'est la personne qui décrit les critères qui existent dans sa base de données pour l'analyse et la classification. Cependant, l'apprentissage non supervisé est autodirigé et regroupe la recherche de données similaires.

1.5.2 Exploration de l'Apprentissage Automatique dans le Renforcement de la Sécurité de l'IdO

L'apprentissage automatique peut être utilisé pour renforcer la sécurité de l'IdO de plusieurs façons. Voici quelques exemples :

◦ Détection d'anomalies : Les modèles d'apprentissage automatique peuvent être utilisés pour détecter les anomalies dans les flux de données de l'IdO. Cela peut inclure des changements dans les schémas de trafic, des fluctuations dans les niveaux de bruit, ou des variations dans les taux d'utilisation des ressources. En apprenant à reconnaître les schémas de données normaux, les modèles peuvent détecter les écarts par rapport à cette norme et signaler une alerte. Cela peut aider à identifier rapidement les problèmes potentiels, tels que des appareils compromis ou des attaques en cours, et permettre une intervention rapide.

◦ Détection d'intrusion : Ces modèles peuvent être utilisés pour détecter les tentatives d'intrusion dans les systèmes IdO. Les modèles peuvent apprendre à reconnaître les schémas de trafic suspects, tels que des tentatives répétées de connexion, des requêtes malveillantes ou des comportements inhabituels. En identifiant ces schémas de trafic, les modèles peuvent signaler une alerte pour signaler une tentative d'intrusion. Cela peut aider à prévenir les attaques avant qu'elles ne réussissent, en fournissant aux équipes de sécurité les informations dont elles ont besoin pour intervenir rapidement.

◦ Détection de comportement malveillant : Les algorithmes d'apprentissage automatisé peuvent également être utilisés pour détecter les comportements malveillants des appareils IdO. Cela peut inclure la reconnaissance des schémas de trafic correspondant à des botnets, des tentatives de DDoS ou d'autres types d'attaques. En détectant ces comportements, les modèles peuvent signaler une alerte pour signaler une activité malveillante. Cela peut aider à prévenir les attaques avant qu'elles ne causent des dommages importants, en permettant aux équipes de sécurité de prendre des mesures préventives.

◦ Prédiction d'incidents : Les techniques de machine learning peuvent être entraînés pour prédire les incidents de sécurité avant qu'ils ne se produisent. En apprenant à reconnaître les signes avant-coureurs des attaques, les modèles peuvent alerter les opérateurs de l'IdO et leur permettre de prendre des mesures préventives avant que l'incident ne se produise. Cela peut inclure la détection des tendances à long terme dans les schémas de trafic, la reconnaissance des comportements suspects ou l'identification des vulnérabilités potentielles. En prédisant les incidents de sécurité, les modèles peuvent aider à protéger les appareils IdO contre les attaques futures.

En supplément, l'apprentissage machine est applicable pour renforcer les contrôles d'accès et évaluer les risques de sécurité des appareils IdO. Les modèles d'apprentissage en profondeur peuvent être utilisés pour identifier les vulnérabilités potentielles des appareils et pour recommander des mesures de sécurité pour les protéger.[8].

1.5.3 Défis de l'utilisation de l'apprentissage automatique pour la sécurité de l'IdO

L'Internet des Objets (IdO) est confronté à des risques de sécurité croissants en raison de sa nature distribuée, de la variété des appareils connectés, et de l'absence de réglementation stricte en matière de sécurité. Pour faire face à ces défis, l'apprentissage automatique est souvent présenté comme une solution potentielle en matière de sécurité de l'IdO. Cependant, cette approche présente des défis importants qui doivent être pris en compte.

L'un des principaux défis de l'utilisation de l'apprentissage automatique pour la sécurité de l'IdO est lié aux risques d'attaques de biais dans les modèles d'apprentissage. Ces attaques peuvent se produire lorsque les modèles d'apprentissage sont entraînés sur des données qui ne représentent pas fidèlement la réalité, ce qui peut entraîner des résultats imprévisibles et potentiellement dangereux. Par exemple, si un modèle d'apprentissage est entraîné sur des données provenant d'un environnement spécifique, il peut ne pas être en mesure de reconnaître les anomalies dans d'autres environnements, ce qui peut entraîner des erreurs de détection de menaces.

Un autre défi important de l'utilisation de l'apprentissage automatique pour la sécurité de l'IdO est la sécurité des modèles d'apprentissage eux-mêmes. Les modèles d'apprentissage automatique peuvent être vulnérables aux attaques, telles que les attaques d'injection de données ou les attaques de perturbation, qui peuvent compromettre l'intégrité des modèles d'apprentissage et perturber leur fonctionnement normal. Il est donc important de surveiller constamment l'activité suspecte et de mettre en place des mécanismes de sécurité appropriés pour protéger les modèles d'apprentissage.

Pour illustrer ces défis, prenons l'exemple d'un système de détection d'intrusion basé sur l'apprentissage automatique. Si le modèle d'apprentissage est entraîné sur des données qui ne représentent pas fidèlement la réalité, il peut être incapable de détecter certaines

anomalies dans les données de l'IdO, ce qui peut entraîner des erreurs de détection de menaces. De même, si le modèle d'apprentissage est compromis par une attaque de perturbation, il peut ne plus être en mesure de détecter correctement les menaces, ce qui peut mettre en danger la sécurité de l'IdO.

L'usage de l'apprentissage automatique pour renforcer la sécurité de l'Internet des Objets (IdO) offre des opportunités prometteuses, mais engendre des défis liés aux risques d'attaques biaisées et à la sécurité des modèles. Des mesures de sécurité, de diversité des données et de sensibilisation sont nécessaires pour maximiser les avantages tout en minimisant les risques[8].

1.6 Approches

1.6.1 Approche 1

Dans cette approche, nous allons explorer en détail l'architecture du système proposé, qui repose sur un système de contrôle qualité basé sur la technologie blockchain. Notre système comprend quatre couches principales : une couche de capteurs IdO, une couche de registre distribué, une couche de contrats intelligents, et enfin, une couche métier abritant diverses fonctions. La technologie blockchain est utilisée pour distribuer en toute sécurité les informations relatives à l'évaluation de la qualité, des actifs, de la logistique et des transactions. Les contrats intelligents définis apportent de l'intelligence, garantissent la confidentialité des données, et automatisent le fonctionnement du système, tandis que les capteurs IdO collectent des données en temps réel. Les modules d'apprentissage automatique sont employés pour le prétraitement et l'analyse des données.

La première couche, celle des capteurs, se sert du GPS pour suivre les mouvements des produits et enregistrer leur position. Les RFID fournissent des informations concernant les transactions, la qualité et les actifs. Lorsque la précision et la quantité de données ne sont pas un enjeu majeur, les codes-barres peuvent être utilisés en raison de leur coût moindre. De plus, d'autres types de capteurs peuvent être exploités pour recueillir des informations supplémentaires, telles que la température ou l'humidité. La deuxième couche est celle du registre distribué, qui englobe quatre aspects principaux de la blockchain : les transactions, les actifs, la logistique et les données de qualité. Chaque acteur de la chaîne d'approvisionnement conserve une copie de ces informations, qu'il s'agisse du fournisseur, du fabricant, du responsable logistique, du détaillant ou de l'opérateur financier. Ces données servent à effectuer un contrôle qualité et à garantir l'efficacité du système. La troisième couche est celle des contrats intelligents, qui sont utilisés pour optimiser l'efficacité de la chaîne d'approvisionnement en collectant et en partageant des données. Afin de préserver la confidentialité, des identités numériques sont mises en place pour contrôler l'accès aux informations. Cette démarche vise à protéger les données confidentielles, notamment entre entreprises concurrentes au sein de la même chaîne d'approvisionnement.

Enfin, la couche métier abrite différentes activités commerciales et offre la possibilité de gérer et de contrôler la qualité, ainsi que de prendre en charge les contrats grâce à la technologie blockchain[9].

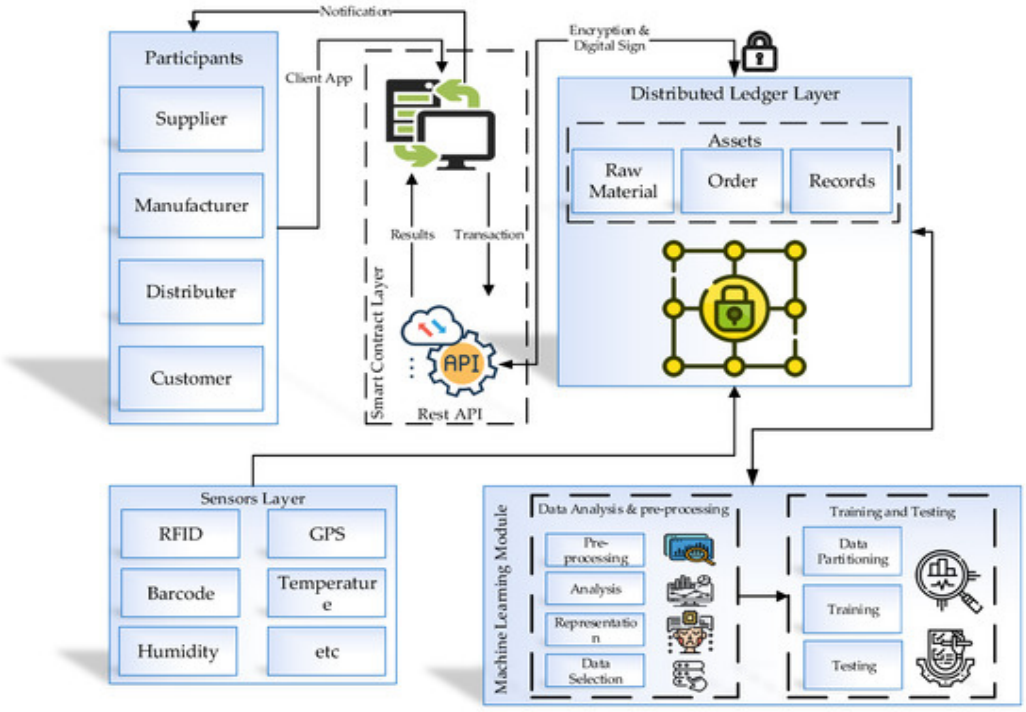


FIG. 1.7 – Architecture du contrôle qualité de la fabrication intelligente basée sur la blockchain.

1.6.2 Approche 2

Cette approche présente en détail le modèle proposé. Les composants développés sont illustrés dans la Figure 1.8. L'initialisation des dispositifs, le calcul de la pertinence à l'aide de l'apprentissage automatique, le hachage numérique et la sécurité basée sur la technologie blockchain sont les principaux éléments du modèle proposé. Le modèle proposé utilise initialement une analyse de régression multiple pour identifier le prochain saut dans le transfert de données agricoles. La régression multiple est une technique statistique couramment utilisée dans de nombreuses applications d'apprentissage automatique pour identifier la relation entre les variables dépendantes et indépendantes. La fonction objective du modèle proposé offre une approche statistique pour analyser le résultat optimal et repose sur divers attributs de réseau.

De plus, la préservation de la vie privée basée sur l'IdO pour la collecte et l'agrégation de données est également essentielle pour réduire les risques liés aux données dans la croissance agricole. Notre mécanisme de sécurité est divisé en trois étapes : capteurs, bords et centres de données. Tout d'abord, les données agricoles sont protégées lors de leur transfert depuis les capteurs vers les dispositifs de bord. Ensuite, le niveau intermédiaire,

composé de différents bords, est protégé contre les comportements suspects. L'intégration de la technologie blockchain offre des fonctionnalités sécurisées de manière distribuée. Dans un tel schéma, les nœuds effectuent des fonctions d'authentification et d'intégrité de manière collaborative sans surcharge excessive. Enfin, les données des dispositifs de bord sont envoyées en toute sécurité vers les centres de données[10].

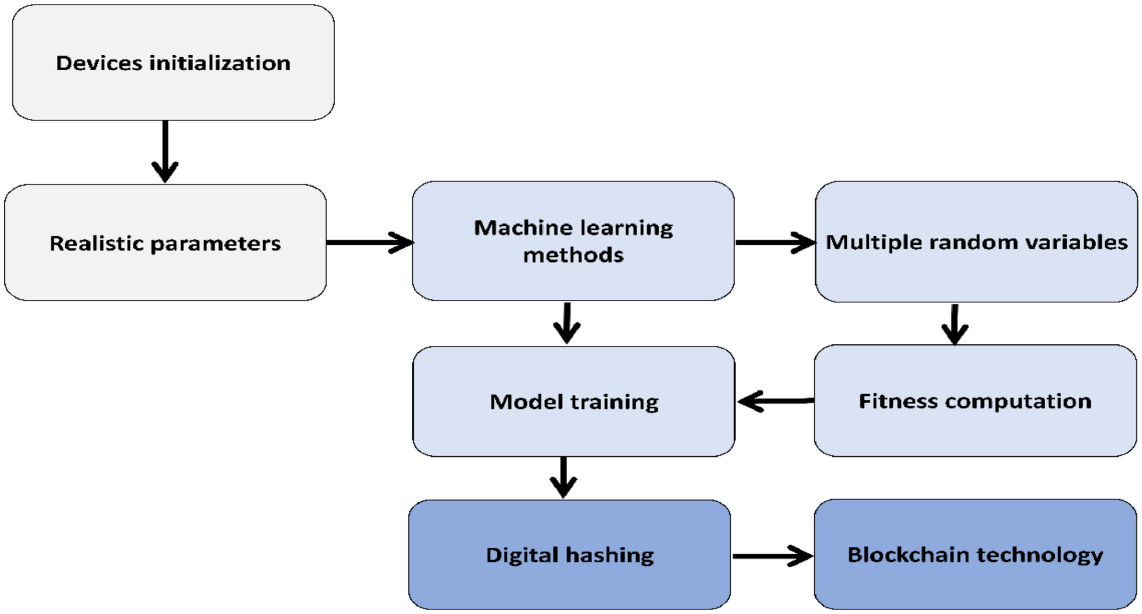


FIG. 1.8 – Schéma fonctionnel illustrant le modèle agricole proposé.

1.6.3 Approche 3

L'architecture BDSDT (Blockchain et apprentissage profond pour la sécurité et la transmission de données) proposée pour les systèmes de soins de santé activés par l'IdO vise à assurer la transmission sécurisée des données entre différentes entités. Ces entités comprennent les dispositifs IdO (Sdi), les serveurs de bord (EDGE) et un vérificateur (V). Le rôle du vérificateur est d'enregistrer toutes les entités participantes avant leur intégration dans le réseau. Les dispositifs IdO, qui englobent des capteurs de pression, de qualité de l'eau et de proximité, sont chargés de la collecte de données, telles que la détection de fuites d'équipement, la surveillance de la qualité de l'eau et la détection de la présence d'objets. Chaque dispositif IdO est connecté à Internet pour la transmission des données. Les serveurs de bord, comprenant des ordinateurs industriels et des serveurs d'analyse de données, traitent les données provenant d'un ou plusieurs dispositifs IdO. L'architecture BDSDT se compose de deux composantes principales : une architecture de sécurité basée sur la blockchain et une architecture de sécurité basée sur l'apprentissage profond. La première enregistre les dispositifs IdO et assure la transmission sécurisée des données, offrant des informations sur l'activité du réseau et la surveillance des performances. Elle protège la source des données, la propriété, les destinations, les itinéraires alternatifs, les mesures de sécurité et protège contre les attaques de falsification des données. Dans l'ar-

chitecture de sécurité basée sur l'apprentissage profond, une technique de l'autoencodeur empilé profond (DSAE) est utilisée pour transformer les données d'origine en un format de dimension réduite, améliorant ainsi la confidentialité des données. L'architecture fonctionne à travers six phases distinctes, comprenant l'initialisation, l'enregistrement, le chiffrement, la création de blocs, la génération de données et le consensus, assurant ainsi une approche globale de la sécurité des données dans le contexte complexe des soins de santé IdO.[11]

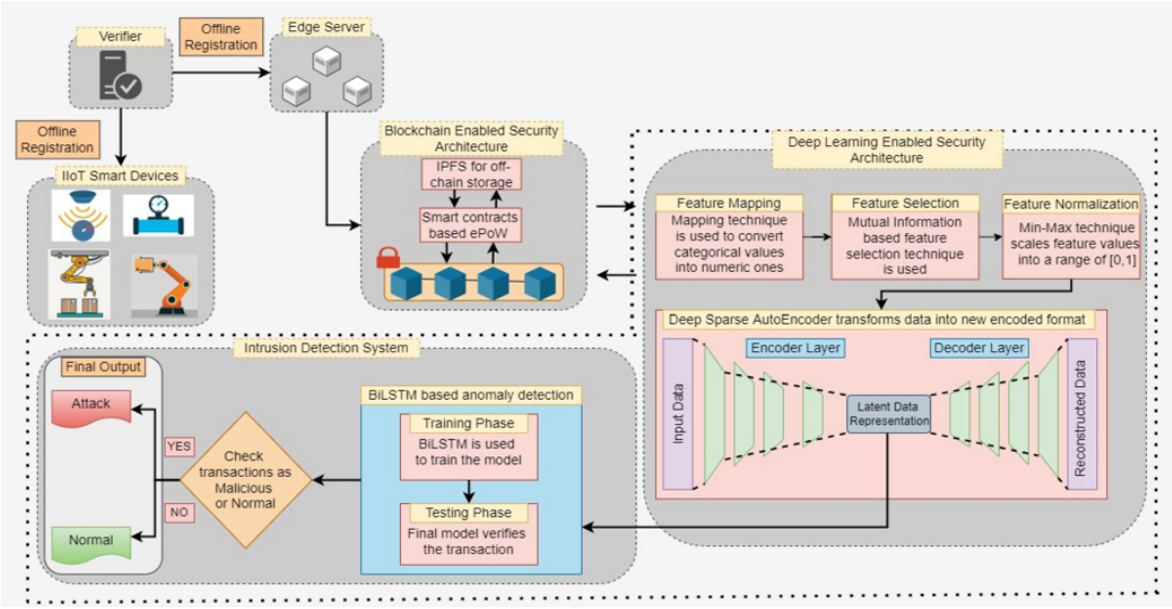


FIG. 1.9 – Approche d'apprentissage profond orchestrée par la blockchain pour sécuriser les données transmission dans l'IdO.

1.6.4 Notre approche

La solution que nous proposons vise à établir un cadre solide pour la détection des intrusions à l'aide d'un modèle d'apprentissage automatique. Ce modèle sera enrichi d'un mécanisme de recyclage périodique afin de maintenir sa précision dans un environnement d'évolution. Pour garantir l'intégrité des données collectées, nous intégrerons également la technologie blockchain. Couche de capteurs : dans cette première couche, les capteurs connectés à l'environnement IdO collectent diverses données provenant des objets surveillés. Ces capteurs transmettent les données collectées sous forme de transactions au réseau. une fois que ces transactions collectées par les capteurs ont été reçues par la couche de contrat intelligent. C'est ici que naissent les contrats intelligents basés sur un modèle pré-entraîné. Ces contrats intelligents, déployés sur la blockchain, Si le modèle de machine learning détecte une anomalie, une alerte est générée pour signaler une éventuelle intrusion puis, Les résultats des contrats intelligents sont soumis vers l'etape suivante ou le mecanisme de consensu s'applique pour notre cas nous proposant l'algorithme Poet en raison de son efficacité énergétique, de son équité, de sa sécurité et de sa capacité d'évo-

lutivité par rapport à d'autres algorithmes comme le Proof of Work (PoW). Cette couche agit comme un grand livre distribué, qui garantit l'intégrité, la sécurité et la traçabilité des données.

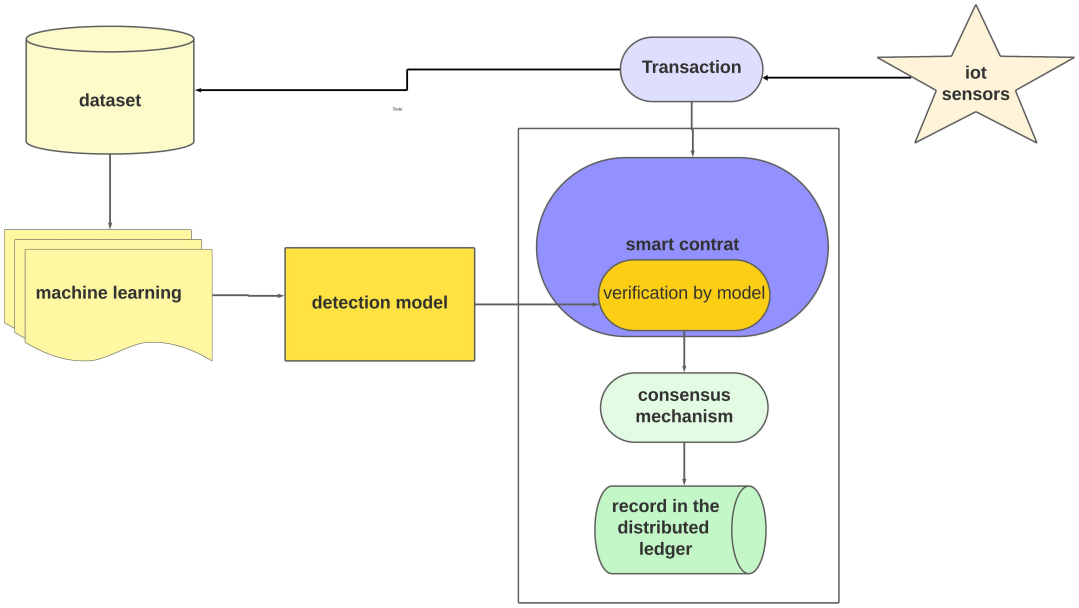


FIG. 1.10 – la structure du modèle proposé.

1.7 Conclusion

Dans ce chapitre, nous avons examiné deux domaines clés de la technologie moderne : Apprentissage automatique pour la sécurité de l'Internet des Objets (IdO) et l'intégration de la blockchain dans l'IdO. L'apprentissage en automatique offre des avantages considérables pour la sécurité de l'IdO en permettant la détection des anomalies, l'identification des comportements malveillants des appareils et la prévention des incidents de sécurité. Cependant, des défis subsistent, notamment en matière de confidentialité des données et de gestion des modèles d'apprentissage en profondeur.

L'intégration de la blockchain dans l'Internet des Objets (IdO) présente des avantages, notamment dans le financement mondial. Malgré des préoccupations en matière de sécurité et de confidentialité, cette fusion offre un potentiel prometteur pour des applications innovantes et sécurisées dans divers secteurs.

Chapitre 2 MÉTHODOLOGIE DE NOTRE RECHERCHE

2.1 Introduction

Dans ce chapitre, nous explorerons en détail les solutions technologiques qui ont été sélectionnées pour répondre aux problématiques soulevées dans cette étude. Notre approche repose sur l'intégration de deux technologies clés : la blockchain et l'apprentissage automatique. La première catégorie de technologies offre une base solide pour stocker et sécuriser les données de manière décentralisée. Elle garantit également leur immuabilité et leur accessibilité transparente. La deuxième catégorie de technologies repose sur des algorithmes d'apprentissage automatique qui joueront un rôle central dans l'analyse des données, la détection d'anomalies et la prise de décision intelligente au sein de notre système.

2.2 Docker

Docker est une plateforme puissante qui simplifie le développement, le déploiement et la gestion d'applications grâce à des conteneurs. Ces derniers offrent un environnement léger et portable pour l'exécution d'applications, ce qui les rend particulièrement adaptés aux systèmes complexes tels que les plateformes blockchain. Docker fonctionne selon une architecture client-serveur, où le client communique avec le démon Docker pour créer, exécuter et distribuer des conteneurs. Le client et le démon peuvent être sur le même système ou connectés à distance via une API REST, des sockets UNIX ou une interface réseau. Docker Compose, un autre client, facilite la gestion d'applications composées de multiples conteneurs[12].

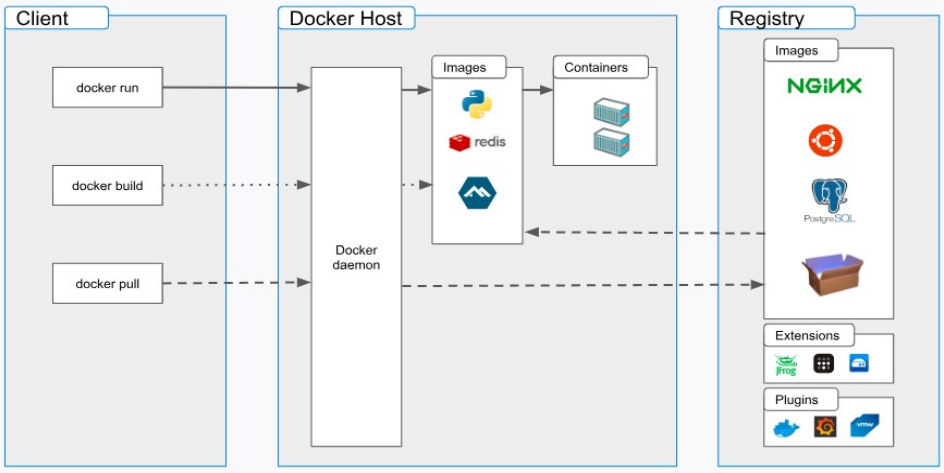


FIG. 2.1 – Architecture du Docker.

- Bureau Docker : Docker Desktop est une application pour Mac, Windows et Linux qui facilite la création et le partage d'applications et de microservices conteneurisés. Il inclut le démon Docker, le client Docker, Docker Compose, Kubernetes, et d'autres outils.
- Démon Docker : Le démon Docker (dockerd) gère les objets Docker tels que les images, les conteneurs, les réseaux et les volumes. Il écoute les requêtes de l'API Docker et peut communiquer avec d'autres démons pour gérer les services.
- Client Docker : Le client Docker (docker) permet aux utilisateurs d'interagir avec Docker. Les commandes telles que docker run sont envoyées au démon dockerd pour exécution. Le client peut se connecter à plusieurs démons.
- Registres Docker : Les registres Docker stockent les images. Docker Hub est public, tandis que vous pouvez gérer un registre privé. Docker extrait les images nécessaires du registre configuré avec les commandes docker pull et docker run, et transfère les images vers le registre avec docker push.
- Objets Docker : Docker implique la création et l'utilisation d'images, de conteneurs, de réseaux, de volumes et d'autres objets.

— Images : Les images sont des modèles de conteneurs basées sur des instructions, les images sont légères grâce aux calques et aux modifications.

— Conteneurs : Un conteneur est une instance exécutable d'une image. Vous pouvez créer, démarrer, arrêter, et connecter des conteneurs aux réseaux et au stockage. Les conteneurs sont relativement isolés, et leur configuration est définie par l'image et les options lors de la création.

2.2.1 Conteneurisation avec Docker pour Sawtooth

Dans le contexte de la blockchain Hyperledger Sawtooth, Docker joue un rôle significatif dans la réalisation de la scalabilité, de la facilité de déploiement et de la cohérence des environnements d'exécution. Voici comment Docker s'intègre à Sawtooth :
- Cohérence Essentielle : Les conteneurs Docker encapsulent l'application et ses dépen-

dances, assurant ainsi une cohérence essentielle. Chaque nœud Sawtooth fonctionne dans un environnement uniforme, évitant les problèmes de compatibilité.
- Scalabilité Optimale : Docker facilite la scalabilité de Sawtooth en utilisant des outils comme Docker Swarm ou Kubernetes pour orchestrer les nœuds en conteneurs. Cela permet d'ajouter facilement des nœuds pour gérer des charges transactionnelles élevées.
- Déploiement Simplifié : Docker simplifie le déploiement des nœuds Sawtooth sur diverses plates-formes. Toutes les configurations sont regroupées dans une image de conteneur, réduisant les risques d'erreurs lors du déploiement.
- Sécurité Par Isolation : Les conteneurs Docker isolent chaque nœud Sawtooth sur le même hôte, renforçant la sécurité et limitant les conséquences potentielles d'une faille de sécurité.
- Développement Plus Rapide : Docker accélère le développement et les tests en créant des environnements isolés. Les développeurs peuvent configurer rapidement des environnements reflétant la production, minimisant les erreurs lors de la transition du code vers la production.

En tirant parti de la technologie de conteneurisation de Docker, Sawtooth devient plus adaptable à divers scénarios de déploiement, en en faisant un choix solide pour les entreprises cherchant à exploiter la puissance de la technologie blockchain.

2.3 Hyperledger Sawtooth

Hyperledger Sawtooth est une plateforme blockchain d'entreprise axée sur la création de réseaux de registres distribués sécurisés, avec une modularité permettant aux entreprises de personnaliser les règles de transaction, les autorisations et les algorithmes de consensus. Elle simplifie le développement d'applications blockchain en séparant la logique d'application du système central, offrant ainsi une solution adaptable aux besoins spécifiques des entreprises.

Dans un réseau Sawtooth, chaque nœud, qu'il s'agisse d'un ordinateur physique, d'une machine virtuelle, d'un conteneur Docker ou d'un pod Kubernetes, joue un rôle spécifique dans le fonctionnement global de la blockchain. Chaque nœud exécute un validateur, une API REST facultative, un moteur de consensus et un ensemble de processeurs de transactions. Ces nœuds Sawtooth utilisent des conteneurs, chacun ayant une fonction définie au sein du réseau. Voici une explication générale de ce que font ces six conteneurs :

1. Un validateur unique utilisant le consensus Devmode
2. Une API REST connectée au validateur
3. Le processeur de transaction Paramètres (sawtooth-settings)
4. Le processeur de transactions IntegerKey (intkey-tp-python)
5. Le processeur de transactions XO (xo-tp-python)
6. Un conteneur client (shell) pour exécuter les commandes Sawtooth

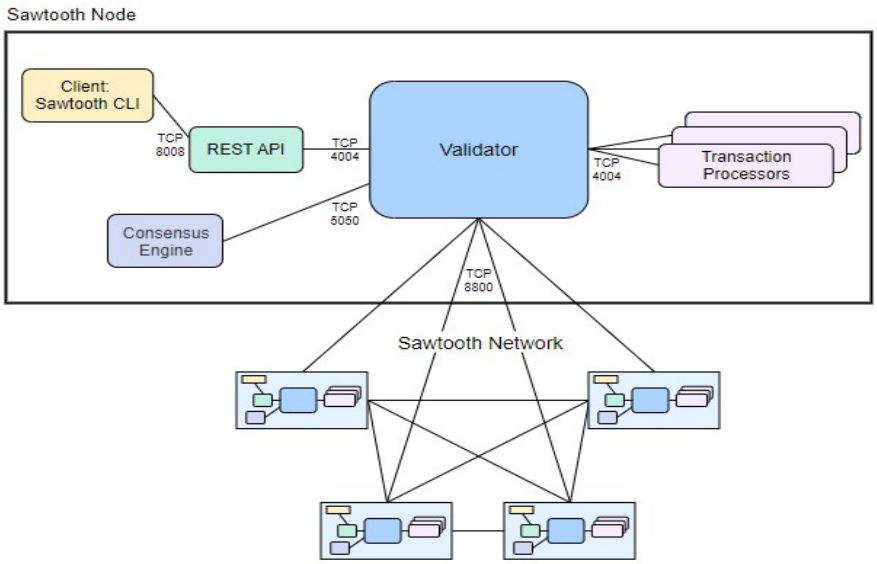


FIG. 2.2 – Exemple d'un réseau de cinq nœuds Sawtooth.

2.3.1 Caractéristiques Distinctives

Sawtooth offre les caractéristiques distinctives suivantes :

- Séparation Application-Système : Clarifie la séparation entre l'application et le système central pour une coexistence fluide des différentes applications.
- Modularité : Personnalise les règles de transaction et de consensus pour s'adapter aux besoins commerciaux.
- Réseaux Privés : Résout les défis des réseaux privés avec des clusters de nœuds distincts pour des permissions isolées.
- Exécution Parallèle : Traite les transactions en parallèle pour des performances améliorées sans compromettre la sécurité.
- Système d'Événements : Soutient la création et la diffusion d'événements liés à la blockchain.
- Compatibilité Ethereum : Intègre Ethereum via Seth pour le déploiement de contrats intelligents EVM.
- Consensus Dynamique : Permet divers algorithmes de consensus, modifiables après la création du réseau, dont Sawtooth PBFT, Sawtooth PoET, Versions PoET, Sawtooth Raft, et Devmode.

2.4 Algorithmes utilisés

2.4.1 CNN

Les CNN sont structurés selon des conceptions différentes par rapport aux réseaux de neurones traditionnels. Chaque couche du réseau neuronal traditionnel est composée d'un

ensemble de neurones qui sont tous liés à tous les neurones de la couche précédente. En revanche, au lieu de neurones entièrement couplés à la couche précédente, chaque couche de CNN n'est liée qu'à un infime pourcentage de neurones. Une structure CNN de base se compose de trois couches : convolution, pooling et une couche entièrement liée . Un filtre ou noyau parcourt l'image d'entrée et crée une conclusion d'un tableau d'entiers dans la couche convolutive. La multiplication du noyau sur une partie de l'entrée produit une valeur unique. En passant le filtre à travers l'ensemble de l'image, plusieurs valeurs sont créées, qui représentent la carte des caractéristiques des données d'entrée. L'utilisation de plusieurs noyaux génère diverses cartes de caractéristiques qui reflètent diverses propriétés des tenseurs d'entrée. L'équation suivante décrit la description mathématique de la convolution couche :

$$M_i = f(M_{i-1} \oplus W_i + b_i)$$

où M_i décrit la carte des caractéristiques de la couche i et $M_0 = X(couched'entrée), W_i$ représente le vecteur de poids du filtre à convolution de la couche i, tandis que b_i et f représentent respectivement le vecteur de biais et la fonction d'activation. La fonction d'activation de l'unité linéaire rectifiée (ReLU) est une fonction non linéaire courante utilisée dans CNN . Le potentiel de CNN réside dans le nombre réduit de paramètres utilisés par rapport au réseau neuronal traditionnel, car il partage le même vecteur de poids et de biais. De plus, contrairement aux classificateurs d'apprentissage automatique traditionnels, il ne nécessite pas d'extraction de fonctionnalités artisanale. La deuxième couche est l'opération de sous-échantillonnage de la couche de pooling et vise à réduire la dimensionnalité de la carte des caractéristiques.
Les opérations de pooling sont classées en deux types : le pooling maximum et le pooling moyen . Pour obtenir les résultats finaux, la sortie finale de la couche de convolution ou d'interrogation est traitée via une ou plusieurs couches entièrement connectées pour les tâches de classification. La couche de sortie finale a le même nombre de nœuds ou de neurones que le nombre de classes de sortie[13].

2.4.1.1 Couche convolutive

Dans la couche convolutive, plusieurs filtres glissent sur la couche pour les données d'entrée données. Une sommation d'une multiplication élément par élément des filtres et du champ récepteur de l'entrée est ensuite calculée lorsque la sortie de cette sommation pondérée de couche est placée en tant qu'élément de la couche suivante. La figure 2 montre que la matrice de filtre (au milieu) est multipliée par la zone focalisée (matrice de gauche), qui est désignée par les couleurs bleu et rouge comme centre. Le résultat de cette multiplication sera stocké à l'endroit correspondant du centre de focus dans le calque suivant. On pourra alors faire glisser la zone de focus et remplir les autres éléments du résultat de convolution.

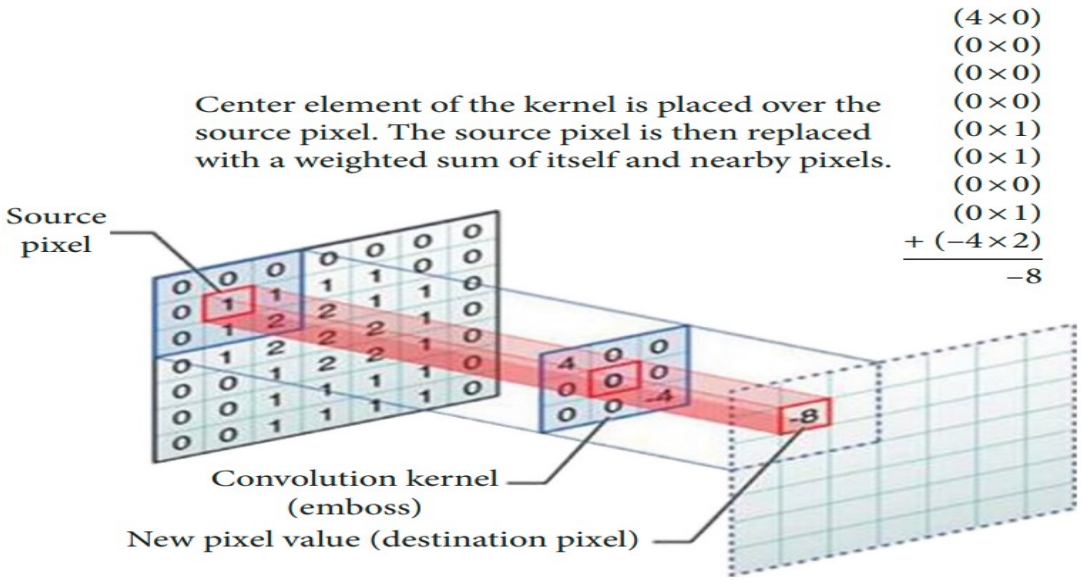


FIG. 2.3 – La couche de convolution.

Chacune des opérations convolutives est spécifiée par la foulée, la taille du filtre et le remplissage nul. La foulée, qui est un nombre entier positif, détermine le pas de glissement. Par exemple, foulée 1 signifie que nous faisons glisser le filtre d'un endroit vers la droite à chaque fois, puis calculons la sortie. La taille du filtre (champ récepteur) doit être fixée sur tous les filtres utilisés dans la même opération convolutive. Le remplissage nul ajoute zéro ligne et colonne à la matrice d'entrée d'origine pour contrôler la taille de la carte des caractéristiques de sortie. Le remplissage zéro vise principalement à inclure les données au bord de la matrice d'entrée. Sans remplissage nul, la sortie de convolution est plus petite que l'entrée. Par conséquent, la taille du réseau diminue en raison de la présence de plusieurs couches de convolutions, ce qui limite le nombre de couches convolutives dans un réseau. Cependant, le remplissage zéro empêche le rétrécissement des réseaux et fournit des couches profondes illimitées dans notre architecture réseau.

2.4.1.2 Non-linéarité

La tâche principale de l'utilisation de la non-linéarité est d'ajuster ou de couper la sortie générée. Plusieurs fonctions non linéaires peuvent être utilisées dans le CNN. Cependant, l'unité linéaire rectifiée (ReLU) est l'une des non-linéarités les plus courantes appliquées dans divers domaines, tels que le traitement d'images. Le ReLU peut être représenté comme

$$\begin{cases} 0, \text{ if } x < 0, \\ x, \text{ if } x \geq 0. \end{cases}$$

2.4.1.3 Couche de Pooling

la couche de pooling réduit grossièrement la dimension des entrées. La méthode de pooling la plus populaire, max pooling, représente la valeur maximale à l'intérieur du filtre de pooling (2×2) comme sortie. D'autres méthodes de regroupement, telles que la moyenne et la sommation, sont disponibles. Cependant, le pooling maximum est une méthode répandue et prometteuse dans la littérature car elle fournit des résultats significatifs en sous-échantillonnant la taille des entrées de 75 % .

2.4.1.4 Couche de Softmax

La couche Softmax est considérée comme une excellente méthode pour démontrer la distribution catégorielle. la fonction softmax, qui est principalement utilisée dans la couche de sortie, est un exposant normalisé des valeurs de sortie. cette fonction est dérivable et représente une certaine probabilité de sortie. De plus, l'élément exponentiel augmente la probabilité de valeur maximale. l'équation softmax est donnée comme suit :

$$O_i = \frac{e^{z_i}}{\sum_{i=1}^{M} e^{z_i}}$$

où o_i est le numéro de sortie softmax i, z_i est la sortie i avant le softmax et M est le nombre total de nœuds de sortie.

2.4.1.5 Prévention du sur-apprentissage et Régularisation

Le surapprentissage est l'un des problèmes graves des techniques d'apprentissage automatique, en particulier dans les modèles complexes comportant un grand nombre de paramètres. Lors du sur-apprentissage, le modèle d'apprentissage fonctionne bien pendant la phase d'entraînement, mais ses performances sont malheureusement médiocres en termes de relativité sur des données inconnues. L'une des approches courantes pour surmonter ce problème consiste à utiliser des techniques de sélection de caractéristiques pour trouver les caractéristiques les plus essentielles des données d'entrée. Ces stratégies peuvent cependant entraîner certaines pertes d'informations utiles. Une autre approche consiste à utiliser des méthodes de régularisation pour réguler la complexité du modèle afin de réduire la pression sur la complexité des paramètres (c'est-à-dire les poids et les biais). Les stratégies de régularisation formalisent les caractéristiques en plaçant une limite sans perte sur l'ampleur des coefficients. Le contrôle des valeurs des paramètres peut réduire le surajustement et améliorer les performances du modèle sur des données inconnues. Par exemple, la technique de régularisation L2 consiste à appliquer une pénalité sur le carré des valeurs du coefficient de pondération. En conséquence, les poids les plus importants deviennent proches de zéro. Nous visons à minimiser la fonction de coût suivante pendant le processus de formation :

$$j(w^1, b, ..., w^L, b^L) = \frac{1}{m} \sum_{i=1}^{m} L(\hat{y}^{(i)}, y^{(i)})$$

où L est la fonction de perte, w est le poids et b est le biais. Désormais, en utilisant la régularisation L2, la fonction de perte deviendra :

$$j(w^1, b, ..., w^L, b^L) = \frac{1}{m} \sum_{i=1}^{m} L(\hat{y}^{(i)}, y^{(i)}) + \frac{\lambda}{2m} \sum_{l=1}^{L} \|w^L\|^2$$

où λ est un paramètre qui peut être ajusté pour contrôler l'effet de régularisation. En utilisant un λ grand, la pénalité de poids sera importante. De même, un petit λ réduira l'effet de régularisation. C'est trivial, car la fonction de coût doit être minimisée. En ajoutant la norme au carré de la matrice de poids et en la multipliant par λ, les poids importants seront réduits afin de minimiser la fonction de coût[13].

2.4.2 Random Forest

Cette section présente une deuxième méthode de machine learning de classification non linéaire.

L'algorithme des forêts aléatoires est une variante du bagging où est agrégé un ensemble d'arbres aléatoires proches de la méthode CART (Breiman et al. ; 1984). Utilisable à la fois en régression et en classification, cet algorithme a montré de très bonnes performances en pratique notamment pour des problèmes complexes (relations non linéaires, interactions, grande dimension, etc.).

2.4.2.1 Arbre de décision

Les arbres de décision, ou arbres aléatoires, sont un ensemble de techniques permettant de construire un classifieur en partitionnant l'espace des observations de façon récursive. La découpe récursive se fait de façon dyadique ce qui donne une structure d'arbre binaire à l'objet final (Figure 2.6). Le premier nœud de l'arbre est la racine et les éléments les plus bas sont les feuilles et constituent la partition de l'espace des observations. Il existe plusieurs algorithmes de construction des arbres aléatoires dont le plus connu est l'algorithme CART (Classification And Regression Trees, Breiman et al. ; 1984). L'algorithme CART procède

À chaque feuille est associée le vote majoritaire des éléments qu'elle contient (-1 pour prédire noir et 1 pour prédire rouge).
En deux étapes pour construire un arbre de décision optimal : la phase d'expansion et la phase d'élagage.

La racine de l'arbre contient toutes les observations $\mathcal{D}_n$. L'algorithme CART recherche la meilleure découpe possible parmi toutes les variables explicatives. Autrement dit, il construit deux sous-parties N_1 et N_2 (les nœuds fils) comme suit :

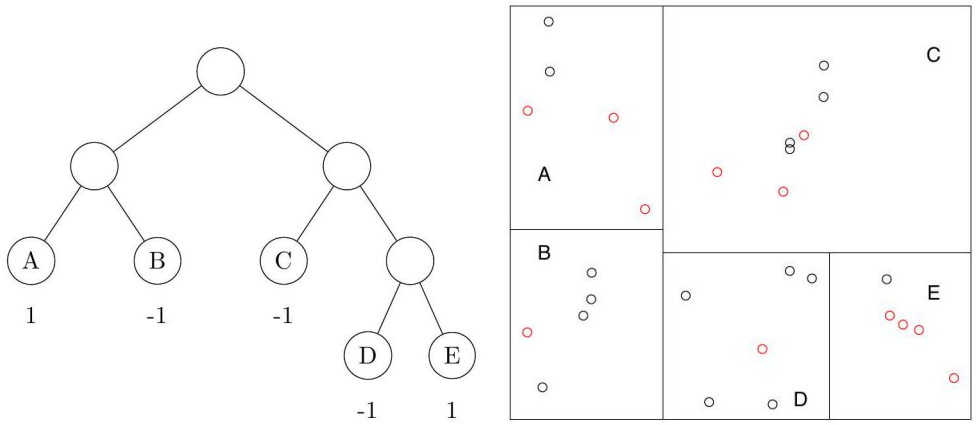


FIG. 2.4 – Classification par arbre.

$$N_1 = \{\mathbf{X}_i, \mathbf{X}_{ij^\star} \leqslant d^\star\},$$
$$N_2 = \{\mathbf{X}_i, \mathbf{X}_{ij^\star} > d^\star\}.$$

Le couple $(j^\star, d^\star)$ est choisi de sorte que chaque nœud fils soit le plus homogène possible. L'homogénéité d'un nœud N se mesure par l'indice de Gini $\sum_{k=1}^{K} \hat{p}_N^k \left(1 - \hat{p}_N^k\right)$, où $\hat{p}_N^k$ est la proportion des éléments de classe k dans le nœud N. L'algorithme CART cherche donc à minimiser l'indice de Gini dans l'ensemble des découpes possibles. Une fois la racine ainsi partitionnée, la procédure est répétée pour chacun de ses fils jusqu'à ce que le chaque nœud ne contienne qu'un seul élément ou bien des observations de même classe. L'arbre ainsi obtenu est appelé arbre maximal (noté $T_{\max}$) et les derniers nœuds construits sont les feuilles. À chaque feuille est associée une prédiction définie par la classe majoritaire des observations qu'elle contient. Le prédicteur de l'arbre est alors l'histogramme des prédictions de chaque feuille, c'est-à-dire la fonction

$$\hat{f}_T(x) = \sum_{m=1}^{|T|} k(m) \mathbb{1}_{x \in N_m},$$

avec $k(m) \in \arg\max_k \hat{p}_{N_m}^k$ la classe majoritaire du nœud N_m et $|T|$ le nombre de feuilles de l'arbre. Notons par ailleurs que l'algorithme CART est adapté à la régression en calculant la moyenne empirique des valeurs de Y_i dans chaque feuille.

L'arbre maximal $T_{\max}$ ainsi construit a l'avantage d'avoir un biais faible, mais il a cependant une variance élevée. C'est pourquoi il existe une seconde étape dans l'algorithme CART pour optimiser les performances d'un arbre maximal en construisant un sous-arbre qui réalise le compromis biais-variance ; c'est la phase d'élagage. Une suite de sous-arbres emboités de l'arbre maximal est construite en minimisant un critère pénalisé. Parmi cette collection d'arbres, l'arbre optimal est celui qui admet les meilleures performances. Il est en général obtenu par validation croisée (voir Gey and Nédélec ; 2005 ainsi que les thèses de Gey ; 2002 et de Tuleau ; 2005 pour un panorama complet sur le sujet)[14].

2.4.2.2 Forêts aléatoires de Breiman

Même si la phase d'élagage améliore les performances d'un arbre de classification en terme de biais et de variance, l'algorithme CART reste une technique instable. En effet, une simple permutation de deux observations de l'ensemble d'apprentissage peut produire un arbre très différent. Les forêts aléatoires de Breiman (2001) permettent de résoudre cette faiblesse de l'algorithme CART et en améliorent les performances. Plus précisément, les forêts aléatoires sont une variante du Bagging (Breiman ; 1996) où la règle de décision est un arbre aléatoire. On construit M arbres aléatoires $\hat{f}_1, \ldots, \hat{f}_M$ sur des échantillons bootstrap $\mathcal{D}_n^1, \ldots, \mathcal{D}_n^M$ contenant des observations tirées aléatoirement (avec ou sans remise) dans $\mathcal{D}_n$. À la différence de CART, un petit nombre de variables est choisi aléatoirement à chaque nœud pour déterminer la meilleure découpe possible. Par défaut, le nombre de variables choisies à chaque nœud est de $\sqrt{p}$ pour la classification et de $p/3$ pour la régression. Les arbres ainsi randomisés sont pleinement développés et ne sont pas élagués ce qui permet la construction d'une collection variée de classifieurs. L'estimateur final est donné par l'agrégation de ces estimateurs, soit le vote majoritaire dans le cas de la classification et la moyenne empirique pour la régression.

Il existe peu de résultats théoriques sur les forêts aléatoires ce qui est notamment du à la complexité de la procédure. C'est pourquoi des études ont été menées dans des cadres simplifiés, voir par exemple Breiman (2004), Biau et al. (2008), Biau (2012), Genuer (2012) et Zhu et al. (2012). Très récemment, Scornet et al. (2014) ont établi un premier résultat de convergence de l'algorithme de Breiman pour le modèle de régression additive.

2.5 Erreur Out-of-bag

Les algorithmes de type bagging - non nécessairement les forêts aléatoires - proposent une estimation de l'erreur de prédiction en tirant parti de l'information apportée par les différents estimateurs agrégés (Breiman ; 1997). Chaque échantillonnage bootstrap laisse de coté un certain nombre d'observations (environ un tiers en pratique) qui sont utilisées pour calculer l'erreur Out-of-bag (OOB), pour Out of bagging ("en dehors du bootstrap").

La procédure d'estimation se formalise comme suit. Soit $\overline{\mathcal{D}}_n^m = \mathcal{D}_n \backslash \mathcal{D}_n^m$ avec $m = 1, \ldots, M$, les échantillons OOB constitués des observations non retenues dans les ensembles bootstrap. Pour chaque donnée $(\mathbf{X}_i, Y_i)$, un classifieur $\hat{f}_i^{oob}$ est construit en agrégeant les arbres ne contenant pas $(\mathbf{X}_i, Y_i)$. Une prédiction est alors donnée par $\hat{Y}_i = \hat{f}_i^{oob}(\mathbf{X}_i)$ et par suite l'erreur OOB de la forêt :

$$\hat{R}^{oob} = \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}_{\hat{Y}_i \neq Y_i}.$$

L'erreur OOB estime l'erreur de classification au même titre que la validation croisée Leave-One-Out au sens où cette erreur est calculée à partir des observations qui ne sont pas utilisées pour l'estimation des $\hat{Y}_i$. L'avantage de cette méthode comparée aux tech-

niques de validation croisée est qu'elle s'effectue durant le processus de construction de la forêt et ne nécessite pas de développements algorithmiques supplémentaires. Néanmoins, il est connu dans la littérature que cette procédure d'estimation de l'erreur est légèrement optimiste, c'est-à-dire qu'elle a tendance à sous évaluer l'erreur de prédiction comme le souligne Breiman (2001). En conséquence, il est préférable d'utiliser des techniques telles que la validation croisée pour estimer l'erreur avec précision. Il est cependant tout à fait envisageable d'utiliser l'erreur OOB pour comparer des classifieurs entre eux comme le fait par exemple Genuer et al. (2010) dans un contexte de sélection de variables.

2.6 Cas de données déséquilibrées

Comme nous l'avons mentionné précédemment, si la proportion des labels positifs est très différente de celle des labels négatifs, les performances de la classification peuvent être altérées. Par exemple, le taux d'erreur peut être très faible pour la classe majoritaire mais à l'inverse très élevé pour la classe minoritaire. Il est donc nécessaire d'adapter les forêts aléatoires pour des données déséquilibrées. Plusieurs approches existent pour résoudre ce problème. La première consiste à forcer l'algorithme à construire des échantillons bootstrap équilibrés soit en sous-échantillonnant la classe majoritaire, soit en sur-échantillonnant les données de la classe minoritaire par un tirage aléatoire avec remise (Chen et al. ; 2004). Sous-échantillonner la classe majoritaire semble une bonne approche si cela n'induit pas de perte d'information ou de la structure des données. C'est la démarche que nous choisissons dans la Section 2.7.

Notons que Chawla et al. (2002) proposent un sur-échantillonnage intelligent à travers une méthode générale qu'ils nomment SMOTE pour Synthetic Minority Over-sampling TEchnique. Au lieu de répliquer les observations minoritaires, ils créent, pour chaque observation minoritaire, de nouvelles données synthétiques dans la direction des k plus proches voisins de même classe. Les auteurs combinent cette technique à un sous-échantillonnage des données majoritaires afin d'améliorer les performances et testent la procédure avec l'algorithme C4.5.

2.7 Conclusion

Ce chapitre méthodologie a exposé les approches et les outils essentiels à la réalisation de notre étude. L'utilisation de la blockchain, de Docker et de Sawtooth offre une base solide pour la gestion des données et la sécurité. Ces technologies garantissent l'intégrité, la transparence et la distribution efficace des informations, créant ainsi un environnement propice à notre recherche.

D'autre part, l'intégration d'algorithmes d'apprentissage automatique tels que CNN et Random Forest permettra d'analyser en profondeur les données collectées, de détecter des motifs, d'identifier des anomalies et d'obtenir des informations pertinentes pour notre

étude. Dans les chapitres à venir, nous montrons et présentons la présentation et à l'analyse des résultats obtenus au cours de notre étude.

Chapitre 3 RÉSULTATS DES TRAVAUX EXPÉRIMENTAUX

3.1 Introduction

Dans ce chapitre, nous présentons les résultats de nos expériences en utilisant les modèles CNN (Convolutional Neural Network) et Random Forest sur le jeu de données NSL-KDD. Aussi dans cette section, nous définissons les métriques et les mesures pour évaluer les performances d'un classificateur en fonction de ses résultats.Nous avons mené ces expérimentations pour évaluer la performance de ces deux approches dans la détection d'anomalies et la classification des intrusions. Les résultats que nous allons partager permettront de mieux comprendre l'efficacité de chaque modèle et d'orienter nos recommandations pour la sécurité des réseaux.

3.2 Ensemble de données NSL-KDD

Les données NSL-KDD proviennent du jeu de données KDD Cup 1999, qui est reconnu pour être largement utilisé dans la recherche liée à la détection d'intrusions. Le jeu de données KDD Cup 1999 a été construit en utilisant des données de trafic réseau simulé, ce qui le rend représentatif des activités réseau courantes. Le jeu de données NSL-KDD est recommandé pour résoudre divers problèmes liés au KDD'99. Pour améliorer la précision, la quantité d'échantillons dans ce jeu de données a été réduite, et le trafic a été filtré. Le Tableau 1 présente le nombre d'échantillons pour les types de trafic normaux et d'attaques dans le jeu de données. Au total, il y a 41 attributs différents. Le jeu de données NSL-KDD est divisé en cinq classes en fonction de l'activité ou des cibles de l'attaquant lors de l'exécution de l'attaque cybernétique.

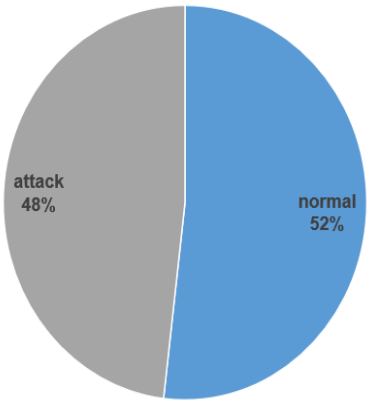


FIG. 3.1 – NSL-KDD dataset.

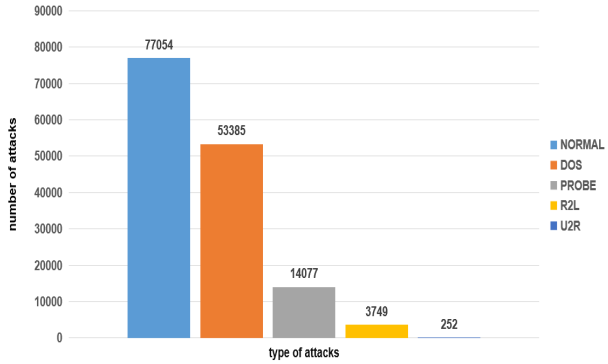


FIG. 3.2 – Type of attacks in NSL-KDD.

3.2.1 Attaques de Déni de Service (DoS)

Attaques de Déni de Service (DoS - Denial of Service) : Cette attaque survient le plus fréquemment dans le jeu de données NSL-KDD. Elle peut être définie comme des cyber-attaques utilisées pour empêcher les utilisateurs normaux de recevoir un service en envoyant plus de demandes de connexion à un serveur que le nombre de demandes qu'il peut gérer, ce qui entraîne l'incapacité du serveur à répondre ou à se mettre hors service pour se protéger. Par exemple, pendant la pandémie de COVID-19, où l'enseignement à distance est devenu plus répandu, des attaques de DoS sont menées pour empêcher les étudiants d'accéder aux examens en ligne dans les établissements d'enseignement supérieur. Bien que des serveurs distribués soient utilisés dans de tels systèmes d'examen, avec le temps, ces attaques affectent l'infrastructure réseau et rendent le système complètement non réactif.

3.2.2 Attaques de Sonde (Probe Attacks)

Attaques de Sonde (Probe Attacks) : Il s'agit d'une cyberattaque visant à recueillir des informations importantes sur un serveur ou toute machine du réseau. Par exemple, avant de pirater un serveur de base de données dans un système d'information de gestion, on tente d'obtenir certaines informations critiques. Les attaques de sonde visent à investiguer des éléments tels que les ports actifs utilisés par le serveur en question, ou des informations cruciales comme l'adresse IP, le système d'exploitation, le type de système de détection d'intrusions (IDS) utilisé.

3.2.3 Attaques R2L

Attaques R2L (Remote-to-Local) : Il s'agit d'une cyberattaque visant à accéder à un ordinateur distant sans autorisation en se faisant passer pour un invité ou un autre utilisateur. Pour obtenir l'accès à l'ordinateur de la victime, diverses méthodes ont été utilisées. Les attaques R2L visent à obtenir un accès local à un système distant. La

machine victime est généralement un ordinateur personnel ou un serveur auquel il est autorisé d'accéder.

3.2.4 Attaques U2R

Attaques U2R (User-to-Root) : Les attaques U2R permettent un accès non autorisé aux serveurs du système cloud. Ces types d'attaques sont généralement réalisés au sein du système réseau où les serveurs sont accessibles, et ils tentent d'acquérir l'autorité d'utilisateur root en exploitant une vulnérabilité ou une faille du serveur. Par exemple, ce sont des attaques où un utilisateur disposant de l'autorisation d'accéder aux serveurs fournissant du contenu en ligne dans le système de gestion de l'université, mais ne disposant pas de privilèges d'administrateur, obtient des privilèges d'administrateur et effectue des opérations non autorisées.

3.3 Métriques d'évaluation

Afin d'analyser et d'évaluer les performances des algorithmes d'apprentissage profond ou de machine learning proposés, nous calculons et vérifions plusieurs mesures pour chacun d'entre eux parmi ceux qui ont été évoqués. Cette démarche vise à offrir une meilleure compréhension et à faciliter la comparaison des performances de chaque algorithme.

3.3.1 Mesures de performance

Les métriques couramment utilisées dans l'évaluation des performances des systèmes de détection d'intrusions (IDS) sont expliquées dans cette section. Tout d'abord, la structure de la matrice de confusion est présentée. Ensuite, nous expliquons brièvement comment les métriques de performance sont calculées. Le Tableau 2 affiche la représentation typique de la matrice de confusion.

Prédit	**Intrusions**	**Normal**	**Total**
Réel Intrusion	TP (Vrai Positif)	FP (Faux Positif)	TP + FP
Réel Normal	FN (Faux Négatif)	TN (Vrai Négatif)	FN + TN
Total	TP + FN	FP + TN	N

Table 3.1 – Structure de la Matrice de Confusion.

La matrice de confusion contient les variables TP (Vrai Positif), TN (Vrai Négatif), FP (Faux Positif) et FN (Faux Négatif), telles que décrites ci-dessous. Une fois que ces variables sont déterminées, nous pouvons calculer les équations allant de 1 à 5 utilisées pour évaluer les performances.

— TP (Vrai Positif) : la quantité d'échantillons positifs correctement classés comme positifs.
— TN (Vrai Négatif) : la quantité d'échantillons négatifs correctement classés comme négatifs.
— FN (Faux Négatif) : la quantité d'échantillons positifs incorrectement classés comme négatifs.
— FP (Faux Positif) : la quantité d'échantillons négatifs incorrectement classés comme positifs.

Une fois que ces valeurs sont connues, elles peuvent être utilisées pour calculer les différentes métriques de performance allant de 1 à 5. Ces métriques sont couramment utilisées pour évaluer l'efficacité des systèmes de détection d'intrusions.

3.3.1.1 Métriques de Performance

1. **Exactitude (Accuracy)** : Il s'agit de la proportion d'échantillons correctement catégorisés par rapport au nombre total d'échantillons. Son calcul est montré dans l'équation 3.1.

$$\text{Exactitude} = \frac{TP + TN}{TP + FP + TN + FN} \tag{3.1}$$

2. **Précision (Precision)** : C'est la proportion du nombre d'échantillons vrais positifs par rapport au nombre total d'échantillons positifs. Son calcul est illustré dans l'équation 3.2.

$$\text{Précision} = \frac{TP}{TP + FP} \tag{3.2}$$

3. **Rappel (Recall)** : Comme indiqué dans l'équation 3.3, il représente la proportion du nombre d'échantillons vrais positifs par rapport à la somme du nombre d'échantillons vrais positifs et du nombre d'échantillons faux négatifs.

$$\text{Rappel} = \frac{TP}{TP + FN} \tag{3.3}$$

4. **Spécificité (Specificity)** : Conformément à l'équation 3.4, c'est la proportion du nombre d'échantillons vrais négatifs par rapport à la somme du nombre d'échantillons vrais négatifs et du nombre d'échantillons faux positifs.

$$\text{Spécificité} = \frac{TN}{TN + FP} \tag{3.4}$$

5. **Score F (F-Score)** : Les valeurs de rappel et de précision sont utilisées pour calculer une nouvelle valeur dans ce critère d'évaluation. Le Score F est calculé en utilisant la formule de la moyenne harmonique, comme indiqué dans l'équation 3.5, où les variables sont les valeurs de précision et de rappel.

$$\text{Score F} = \frac{2 \cdot \text{Rappel} \cdot \text{Précision}}{\text{Rappel} + \text{Précision}} \tag{3.5}$$

3.4 Analyse des Étapes du Code de Random Forest

Dans cette section, nous allons examiner en détail chaque étape du code d'exploration et de modélisation des données du jeu de données NSL-KDD. Ces étapes sont cruciales pour comprendre comment le code traite les données et construit le modèle de classification des attaques réseau.

3.4.1 Importation des Bibliothèques

La première étape consiste à importer les bibliothèques nécessaires au traitement des données et à la modélisation. Les bibliothèques incluent NumPy pour les calculs numériques, pandas pour la manipulation des données, Matplotlib et Seaborn pour la visualisation, ainsi que des modules spécifiques de scikit-learn pour les modèles de machine learning. Ces bibliothèques sont essentielles pour effectuer des analyses avancées et créer des modèles.

3.4.2 Extraction des Données

Après l'importation des bibliothèques, le code extrait les données à partir de fichiers spécifiques. Deux fichiers sont chargés : le jeu de données d'entraînement complet et le jeu de données de test. Cette étape est cruciale car elle fournit les données brutes sur lesquelles nous allons travailler.

3.4.3 Ajout des Noms de Colonnes

Le jeu de données NSL-KDD ne comprend pas de noms de colonnes explicites. Par conséquent, cette étape ajoute des noms de colonnes appropriés aux données. L'ajout de noms de colonnes facilite la compréhension des données et permet de référencer les caractéristiques de manière claire.

3.4.4 Transformation des Données

Une partie importante du code consiste à transformer les données brutes en une forme adaptée à l'analyse et à la modélisation. Les attaques sont classées en deux catégories : "normal" et "attaque". De plus, des sous-catégories d'attaques telles que DoS, Probe, U2R (escalade de privilèges) et R2L (accès à distance) sont définies. Cette étape permet de créer des labels pour les données, ce qui est essentiel pour la classification. De plus, des statistiques sont extraites des données pour fournir des informations contextuelles.

n-estimators	random-stat	min-samples-split
100	0	2

TABLE 3.2 – Les hyper-paramètres du modèle Random Forest.

3.4.5 Ingénierie des Caractéristiques

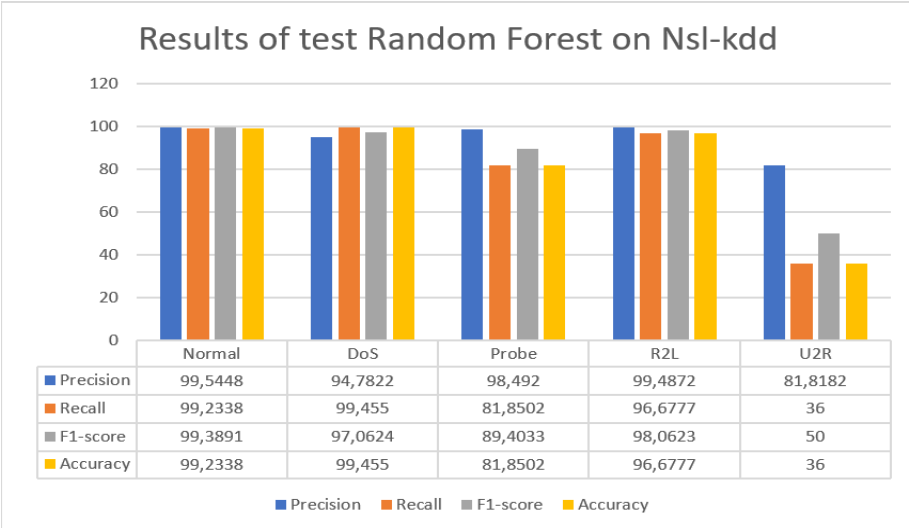
Une autre étape cruciale est l'ingénierie des caractéristiques, où les données sont préparées pour l'entraînement du modèle. Les caractéristiques catégorielles telles que 'protocol-type', 'service' et 'flag' sont encodées en utilisant la technique de "one-hot encoding". Cela permet de traiter les caractéristiques catégorielles dans le modèle. Les caractéristiques numériques telles que 'duration', 'src-bytes' et 'dst-bytes' sont incluses telles quelles. Cette étape prépare les données pour être utilisées dans le modèle de classification.

3.4.6 Entraînement du Modèle RandomForestClassifier pour la Classification Multi-Classe

Le code crée et entraîne un modèle RandomForestClassifier pour effectuer la classification multi-classe des attaques réseau. Ce modèle est ajusté aux données d'entraînement afin d'apprendre les modèles de classification des attaques. Il s'agit d'une étape essentielle dans la construction du modèle.

3.4.7 Affichage des Résultats

La performance du modèle est évaluée à l'aide de diverses métriques telles que la matrice de confusion, le rapport de classification et l'accuracy. Ces métriques permettent d'évaluer à quel point le modèle est performant dans la classification des attaques réseau. Les résultats sont affichés pour fournir une évaluation claire de la performance du modèle.



	Normal	DoS	Probe	R2L	U2R
Precision	99,5448	94,7822	98,492	99,4872	81,8182
Recall	99,2338	99,455	81,8502	96,6777	36
F1-score	99,3891	97,0624	89,4033	98,0623	50
Accuracy	99,2338	99,455	81,8502	96,6777	36

FIG. 3.3 – Resultats de test Random Forest sur Nsl-kdd.

Les résultats du test de notre modèle Random Forest sur le jeu de données NSL-KDD sont les suivants :

1. Normal : Notre modèle Random Forest a obtenu une précision impressionnante avec un recall de 99,23% pour la classe "normal". Cela signifie que le modèle a correctement classé la grande majorité des données normales sans erreur.

2. DoS (Déni de Service) : La classe "dos" a également montré une excellente performance, avec un recall de 99,45%. Cela indique que le modèle a bien réussi à identifier les attaques de type DoS dans le jeu de données.

3. Probe : Recall pour la classe "probe" est légèrement plus bas, à 81,85%. Cela signifie que le modèle a correctement identifié les tentatives de sondage ou d'exploration dans environ 81,85% des cas. Cette classe peut être plus difficile à détecter en raison de ses caractéristiques.

4. R2L (Root to Local) : La classe "r2l" a obtenu un recall de 96,67%. Le modèle a réussi à détecter avec précision les tentatives d'accès illégal de type R2L dans la grande majorité des cas.

5. U2R (User to Root) : Recall le plus bas a été obtenue pour la classe "u2r", à 36%. Cette classe est souvent difficile à détecter en raison de son nombre limité d'exemples dans le jeu de données.

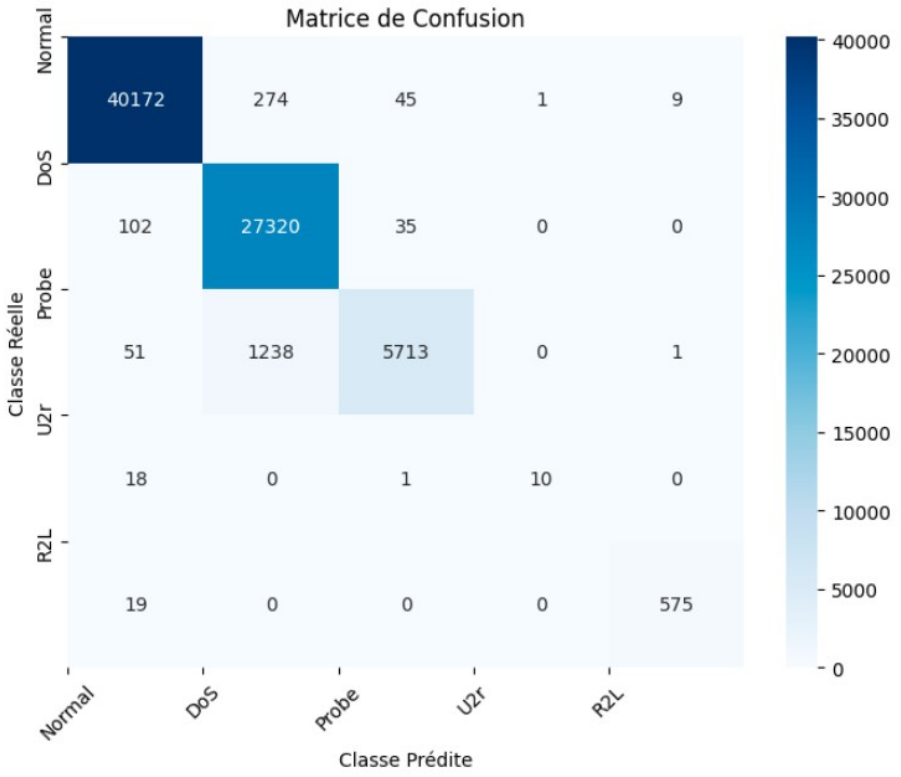


FIG. 3.4 – Matrice de confusion de Random Forest sur Nsl-kdd.

L'évaluation générale du modèle basée sur les résultats de la matrice de confusion indique une performance globalement solide. Le modèle a bien réussi à classer les échantillons

des classes "normal" et "dos", avec un nombre élevé de prédictions correctes, représenté par les valeurs élevées sur la diagonale principale de la matrice.
Le modèle a des difficultés avec la classe "probe", comme indiqué par la valeur relativement faible sur la diagonale principale pour cette classe. Il a également du mal avec les classes "r2l" et "u2r", où les valeurs sur la diagonale principale sont encore plus faibles. Il est important de noter que ces classes ont des volumes de données plus petits, ce qui peut rendre la tâche de classification plus difficile.

3.5 Analyse des Étapes du Code de CNN

3.5.1 Importation des bibliothèques

Le code commence par l'importation de nombreuses bibliothèques Python, notamment celles pour le traitement des données, l'apprentissage automatique (Scikit-Learn), le deep learning (Keras), la visualisation, etc.

3.5.2 Chargement des données

Les données d'entraînement et de test sont chargées à partir de fichiers CSV à l'aide de pandas, puis stockées respectivement dans deux dataframes distincts.

3.5.3 Prétraitement des données

Les colonnes de `difficulty_level` sont supprimées des deux dataframes car elles ne sont pas nécessaires. Les colonnes catégorielles (telles que `protocol_type`, `service`, et `flag`) sont converties en encodage one-hot. Cela permet de transformer des variables catégorielles en variables binaires pour être utilisées dans les modèles d'apprentissage automatique. Les données sont normalisées (mise à l'échelle) pour s'assurer qu'elles sont dans la même plage.

3.5.4 Création de la colonne de classe

Une colonne de classe appelée "Class" est créée en fonction des valeurs des colonnes de sous-classe. Les classes comprennent "Normal", "DOS" (Denial of Service), "Probe", "R2L" (Remote to Local), et "U2R" (User to Root).

3.5.5 Exploration des données

Le code affiche le nombre d'occurrences de chaque classe dans la colonne "Class" à l'aide de la méthode `value_counts()`.

3.5.6 Séparation des données en caractéristiques et cibles

Les données sont divisées en caractéristiques (`combined_data_X`) et en cible (`y_train`), où "X" contient toutes les caractéristiques et "y" contient les étiquettes de classe.

3.5.7 Entraînement du modèle

Le modèle d'apprentissage automatique est entraîné à l'aide des données d'entraînement (`combined_data_X` et `y_train`). Différents algorithmes d'apprentissage automatique peuvent être utilisés, tels que les machines à vecteurs de support (SVM), les arbres de décision, ou les réseaux de neurones profonds, en fonction des besoins du problème.

3.5.7.1 Les hyper-paramètres du modèle CNN

Batch size	number of epochs	optimizor method
32	10	adam

TABLE 3.3 – Les hyper-paramètres du modèle CNN.

3.5.7.2 Architecture du modèle CNN

Cette architecture de réseau neuronal convolutif (CNN) est conçue pour les tâches de classification multi-classes. Il prend en entrée des données de forme (122, 1), qui représentent la séquence de caractéristiques. L'architecture comprend une couche convolutive 1D avec 64 filtres et une taille de noyau de 122, permettant au modèle d'apprendre des modèles importants dans les données. Une couche de pooling maximal réduit les dimensions spatiales, suivie d'une normalisation par lots pour la stabilisation.

Ensuite, les données sont aplaties et transmises via une couche d'abandon avec un taux de 50 % pour éviter le surajustement. Le modèle traverse ensuite une couche dense de 5 unités, qui est suivie d'une activation softmax pour générer des probabilités de classe pour les cinq classes cibles. Le CNN vise à classer avec précision les séquences d'entrée dans l'une de ces classes en fonction des fonctionnalités apprises.

Le modèle est entraîné à l'aide de la perte « categorical-crossentropy » et de l'optimiseur « adam ». Cette architecture exploite les atouts des CNN pour capturer automatiquement les caractéristiques hiérarchiques, ce qui la rend efficace pour les tâches impliquant des séquences comme la nôtre.

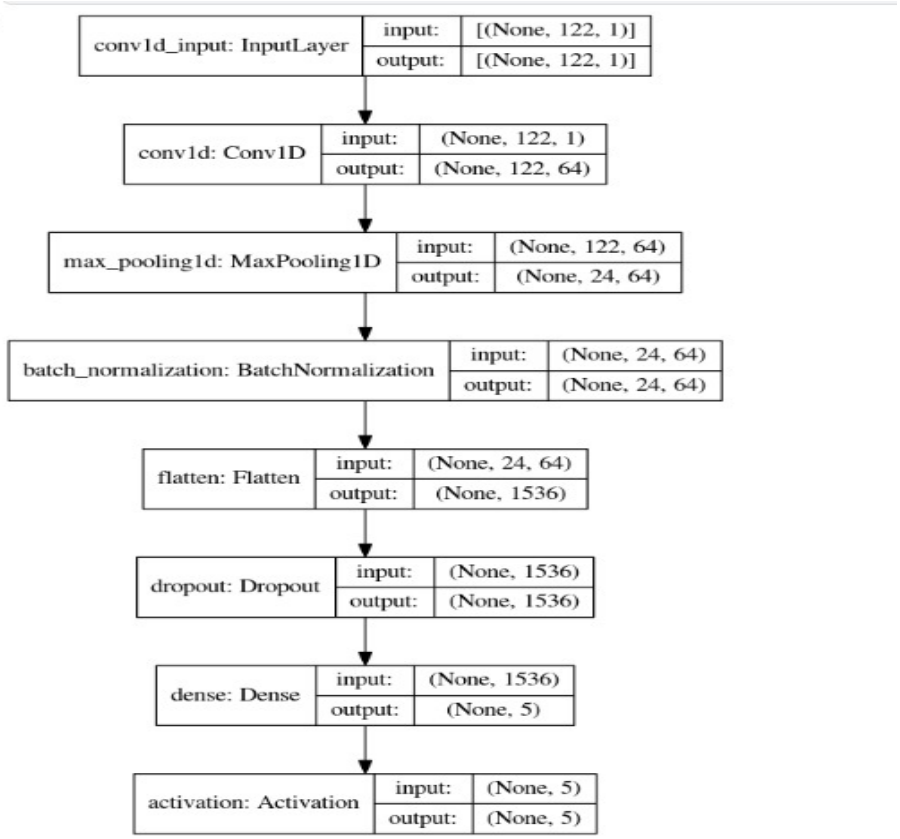


FIG. 3.5 – Architecture du modèle CNN.

3.5.8 Évaluation du modèle

Une fois le modèle entraîné, il est évalué à l'aide des données de test (`qp`) pour mesurer ses performances. Les métriques couramment utilisées incluent la précision, le rappel, la F-mesure et la matrice de confusion.

3.5.9 Interprétation des résultats

Les résultats de l'évaluation du modèle sont interprétés pour déterminer sa capacité à classer correctement les données de test en fonction des classes prédites. Cela permet de mesurer l'efficacité du modèle dans la détection des anomalies ou des intrusions.

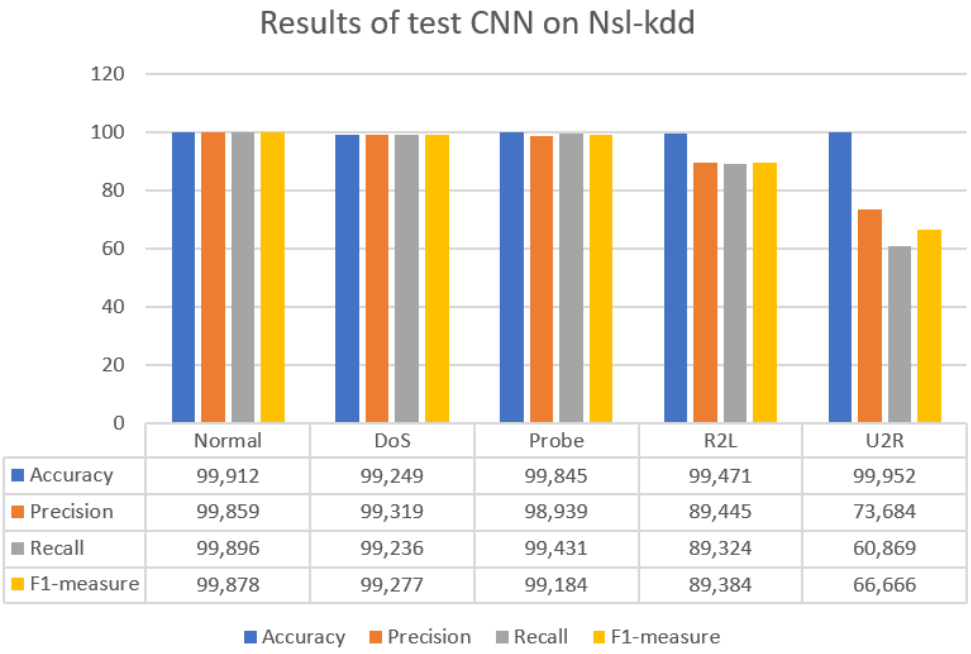


FIG. 3.6 – Resultats de test CNN sur Nsl-kdd.

Les résultats du test de notre modèle CNN sur le jeu de données NSL-KDD sont les suivants :

1. Normal : Le modèle a atteint une précision de 99,85 % dans la détection des connexions normales. Cela signifie que le modèle est extrêmement performant pour identifier les connexions qui ne présentent pas de comportement malveillant. Une précision élevée dans cette catégorie est cruciale pour éviter de fausses alertes sur les activités légitimes.

2. DoS (Déni de Service) : Le modèle a obtenu une précision de 99,31 % dans la détection des attaques de type DoS. Cela indique que le modèle est très efficace pour reconnaître les tentatives de surcharge de système visant à rendre les services indisponibles.

3. Probe : Pour les activités de type Probe (sondage), le modèle a atteint une précision de 99,93 %. Cette catégorie concerne généralement les tentatives d'exploration du réseau pour identifier des vulnérabilités. Le modèle est donc capable de repérer ces comportements potentiellement suspects avec une grande précision.

4. R2L (Accès Non Autorisé depuis un Réseau Local à Distance) : Dans cette catégorie, le modèle présente une précision légèrement inférieure de 89,44 %. Les attaques de type R2L consistent en des tentatives d'accès non autorisé depuis un réseau local distant. La raison derrière les mesures inférieures pour R2L et U2R réside dans le nombre limité d'attaques de ces types dans le jeu de données, ce qui rend plus difficile leur détection. Bien que la précision soit un peu moins élevée, elle reste significative pour détecter ces intrusions.

5. U2R (Accès Non Autorisé à un Utilisateur) : Le modèle a obtenu une précision de 73,68 % dans la catégorie U2R, qui traite des tentatives d'accès non autorisé aux privilèges d'un utilisateur. Comme pour la catégorie R2L, la précision légèrement plus faible s'explique par le nombre limité d'attaques de ce type dans le jeu de données. Malgré cela, cette

précision demeure importante pour identifier de telles attaques.

En résumé, notre modèle CNN s'avère performant pour détecter les activités normales et diverses attaques, y compris les attaques de type DoS et Probe. Néanmoins, sa précision peut être légèrement inférieure pour les attaques de type R2L et U2R, en raison de leur rareté dans les données. Malgré cela, il reste efficace pour repérer ces comportements malveillants, démontrant ainsi son efficacité dans la détection des menaces sur le réseau NSL-KDD.

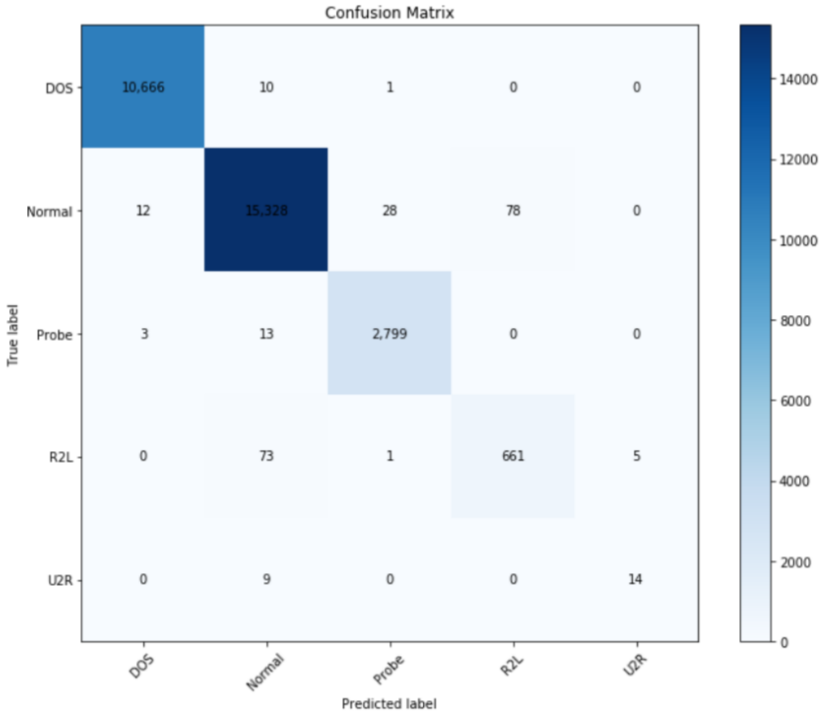


FIG. 3.7 – Matrice de confusion de CNN sur Nsl-kdd.

La matrice de confusion révèle la performance du modèle de classification pour cinq classes : "normal", "dos", "probe", "r2l", et "u2r". Les nombres sur la diagonale principale représentent les prédictions correctes, tandis que les nombres en dehors de la diagonale indiquent les erreurs de classification. Le modèle a bien fonctionné pour les classes "normal", "dos", et "probe", avec un taux de précision élevé. Cependant, il a eu plus de difficulté avec les classes "r2l" et "u2r", en raison du nombre limité d'exemples disponibles pour ces classes, ce qui a entraîné des erreurs de classification plus fréquentes pour ces catégories spécifiques.

3.6 Comparaison entre les résultats du modèle Random Forest et du modèle CNN

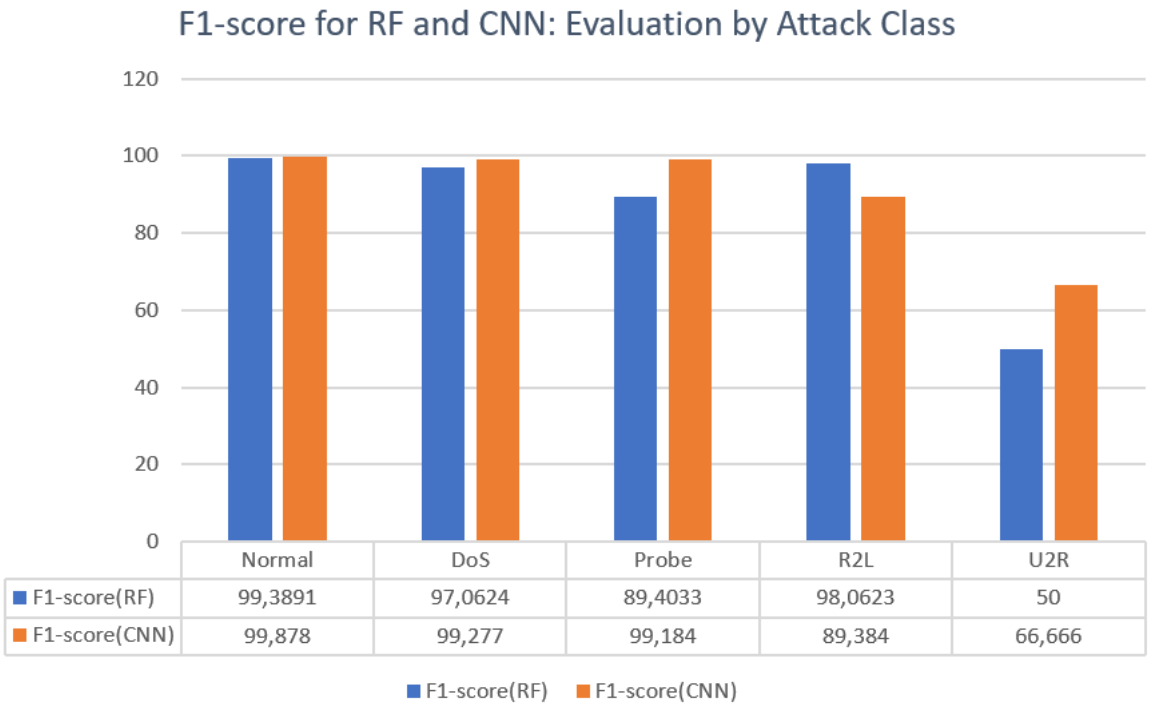


FIG. 3.8 – Les mesures du F1-score pour les deux modèles CNN et Random Forest.

En comparant les performances des deux modèles, il est clair que le modèle CNN a obtenu des scores F1 plus élevés dans l'ensemble pour toutes les classes, à l'exception de la classe "U2R".

Le modèle CNN a particulièrement bien performé pour les classes "normal", "DoS" et "probe", avec des scores F1 élevés dépassant 99

Cependant, le modèle Random Forest a surpassé le modèle CNN pour la classe "R2L", obtenant un score F1 de 98,06

En ce qui concerne la classe "U2R", le modèle CNN a montré des performances relativement plus faibles avec un score F1 de 66,66

En résumé, le modèle CNN a généralement surpassé le modèle Random Forest en termes de scores F1 pour la plupart des classes, témoignant de sa capacité à mieux détecter les intrusions dans le jeu de données NSL KDD. Cependant, il est important de noter que les performances peuvent varier en fonction de la classe, et il peut être judicieux d'envisager une combinaison de modèles ou des ajustements spécifiques pour améliorer encore la détection des intrusions.

3.7 Conclusion

En conclusion, ce chapitre a mis en lumière les performances de nos modèles CNN et Random Forest dans la classification d'intrusions sur le jeu de données NSL-KDD. Nous avons observé que chaque modèle a ses avantages et ses limites, et que le choix du modèle dépendra des objectifs spécifiques de la détection d'intrusions. Le CNN a montré une grande capacité à extraire des caractéristiques importantes à partir de données complexes, tandis que le Random Forest s'est avéré robuste et acceptable et interprétable. Ces résultats fournissent une base solide pour la prise de décision quant à l'approche à adopter pour renforcer la sécurité de notre réseau.

Conclusion générale

Cette étude a examiné en profondeur l'intégration de l'apprentissage automatique et de la Blockchain pour renforcer la sécurité dans l'Internet des Objets (IdO) en abordant les défis liés à la confidentialité, la fiabilité et la sécurité des données. L'utilisation combinée de l'apprentissage automatique et de la Blockchain présente un potentiel considérable pour résoudre ces problématiques. Nous avons examiné comment l'apprentissage automatique peut créer une "Intelligence des Objets" et comment la Blockchain peut assurer la sécurité des données de manière inaltérable. Ces deux technologies, lorsqu'elles sont appliquées de manière appropriée, offrent des solutions prometteuses pour renforcer la sécurité des réseaux IdO et préserver la confidentialité des données sensibles.

Quant aux perspectives futures, plusieurs avenues sont à explorer. Il est essentiel d'optimiser davantage les algorithmes d'apprentissage automatique pour les adapter aux contraintes particulières de l'IdO, notamment les ressources limitées des dispositifs. De plus, la recherche devrait se concentrer sur le développement de cas d'utilisation concrets démontrant l'efficacité de l'intégration de l'apprentissage automatique et de la Blockchain dans des domaines tels que la santé, la logistique, l'industrie, et bien d'autres. À mesure que la puissance de calcul des ordinateurs quantiques augmente, il est nécessaire de se pencher sur la sécurité quantique dans le contexte de la Blockchain. En outre, il est crucial de développer des normes et des réglementations pour encadrer l'utilisation éthique de ces technologies en constante évolution dans les réseaux IdO. Enfin, les questions éthiques liées à la vie privée et à la sécurité des données dans le contexte de l'intelligence artificielle et de la Blockchain devront être explorées plus en profondeur pour élaborer des cadres éthiques solides. En résumé, l'intégration de l'apprentissage automatique et de la Blockchain promet d'améliorer significativement la sécurité de l'IdO, et les futures recherches dans ce domaine devraient contribuer à façonner un avenir plus sûr et plus efficace pour l'IdO.

Bibliographie

[1] *Internet of Things-IOT : Definition, Characteristics, Architecture, Enabling Technologies, Application Future Challenges*. http://www.opjstamnar.com/download/Worksheet/Day-110/IP-XI.pdf/. 09/04/2023. 2023.

[2] *architecture IoT à quatre couches*. https://iotindustriel.com/iot-iiot/architecture-iot-lessentiel-a-savoir/. 09/04/2023. le 29 October 2023.

[3] *Défis de sécurité de l'Internet des Objets Problèmes et solutions*. https://www.ummto.dz/dspace/bitstream/handle/ummto/12608/LarrasMelissa.pdf?sequence=1&isAllowed=y. 13/05/2023. 2023.

[4] *What Is IoT Security ?* https://www.proofpoint.com/uk/threat-reference/iot-security. 10/05/2023. 2023.

[5] *Big IoT Data Analytics : Architecture, Opportunities, and Open Research Challenges - Scientific Figure on ResearchGate.* https://www.researchgate.net/figure/IoT-architecture-and-big-data-analytics-A-novel-meta-model-based-approach-for-integrating_fig3_316240052. 13/05/2023. 2023.

[6] *deep learning and iot for Sale*. https://images.app.goo.gl/c1HUoXBEJ13ejdNYA. 10/05/2023. 2023.

[7] *Introduction à l'apprentissage automatique*. https://eduscol.education.fr/sti/sites/eduscol.education.fr.sti/files/ressources/pedagogiques/14512/14512-introduction-lapprentissage-automatique-ensps.pdf. 20/07/2023. 2023.

[8] Mohamed Ahzam Amanullah et al. "Deep learning and big data technologies for IoT security". In : *Computer Communications* 151 (2020), p. 495-517.

[9] Zeinab Shahbazi et Yung-Cheol Byun. "Integration of blockchain, IoT and machine learning for multistage quality control and enhancing security in smart manufacturing". In : *Sensors* 21.4 (2021), p. 1467.

[10] Tanzila Saba et al. "Trust-based decentralized blockchain system with machine learning using Internet of agriculture things". In : *Computers and Electrical Engineering* 108 (2023), p. 108674.

[11] Prabhat KUMAR et al. “A blockchain-orchestrated deep learning approach for secure data transmission in IoT-enabled healthcare system”. In : *Journal of Parallel and Distributed Computing* 172 (2023), p. 69-83.

[12] *Docker*. https://www.docker.com/. 20/07/2023. 2023.

[13] *Social Touch Gesture Recognition Using Convolutional Neural Network*. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6197001/. 20/07/2023. 2023.

[14] *ArbresCARTetForêtsaléatoires,Importanceet sélectiondevariables*. chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://hal.science/hal-01387654v2/document. 20/07/2023. 2023.